

08-01-00

A



SKJERVEN
MORRILL
MACPHERSON

Docket No.: M-8368 U

July 31, 2000

Box Patent Application
Commissioner For Patents
Washington, D. C. 20231

Enclosed herewith for filing is a patent application, as follows:

Inventor(s): Jain, Bijendra N.; McCloghrie, Keith
Title: Method And Apparatus For Estimating Delay And Jitter Between Many Network Routers
Using Measurements Between A Preferred Set Of Routers
X Return Receipt Postcard
X This Transmittal Letter (in duplicate)
53 page(s) Specification (not including claims)
18 page(s) Claims
1 page Abstract
7 Sheet(s) of Drawings
3 page(s) Declaration For Patent Application and Power of Attorney (Unexecuted)

JC777 U.S. PTO
09/638180
07/31/00

CLAIMS AS FILED

For	Number Filed			Number Extra		Rate		Basic Fee
Total Claims	48	-20	=	28	x	\$18.00	=	\$ 504.00
Independent Claims	4	-3	=	1	x	\$78.00	=	\$ 78.00
<input type="checkbox"/>	Fee of _____ for the first filing of one or more multiple dependent claims per application							\$
<input type="checkbox"/>	Fee for Request for Extension of Time							\$

Please make the following charges to Deposit Account 19-2386:

- ☒ Total fee for filing the patent application in the amount of \$ 1,272.00
- ☒ The Commissioner is hereby authorized to charge any additional fees which may be required, or credit any overpayment to Deposit Account 19-2386.

EXPRESS MAIL LABEL NO:
EL563589105US

Respectfully submitted,

Sam Campbell
Attorney for Applicant(s)
Reg. No. 42,381

"Express Mail" mailing label number:

EL563589105US

**METHOD AND APPARATUS FOR ESTIMATING DELAY AND JITTER
BETWEEN MANY NETWORK ROUTERS USING MEASUREMENTS
BETWEEN A PREFERRED SET OF ROUTERS**

Jain, Bijendra N.
McCloghrie, Keith

5

CROSS-REFERENCES TO RELATED APPLICATIONS

10 This application is related to patent Application No. 09/583,177 entitled
"METHOD AND APPARATUS FOR ESTIMATING DELAY AND JITTER
BETWEEN NETWORK ROUTERS," having B. N. Jain and K. McCloghrie as
inventors. This application is assigned to Cisco Technology, Inc., the assignee of the
present invention, and is hereby incorporated by reference herein.

BACKGROUND OF THE INVENTION

Field of the Invention

15 The present invention relates to computer networks, and, more particularly, to
a method and apparatus for estimating a network performance metric, such as delay
and delay-jitter (jitter), between a collection of pairs of network elements (e.g.,
routers) in a network.

Description of the Related Art

20 As computer networks grow, in terms of the number of network elements (e.g.,
routers) contained therein, the measurement of network performance metrics becomes
of increasing importance. By measuring such metrics, network parameters can be
tuned in order to provide optimal performance. Moreover, the network's architecture
can be adjusted and growth planned to allow the network to grow in a controllable
25 fashion. One such metric is the delay experienced by data packets flowing between
certain of a network's routers (i.e., travel time between the routers). Another is the

jitter, or deviation in delay, experienced by such data packets. Thus, there is a growing need to continuously monitor network delay and jitter between multiple pairs of routers in a network such as an enterprise or service-provider network. In service-provider networks, particularly, such performance monitoring is needed in order to
5 verify service-level agreements between a service provider and customers.

Unfortunately, current methods of monitoring are not as useful as might be desired. For example, one current method for monitoring network delay and jitter requires the measurement of delay and jitter between every specified pair of routers by exchanging probe packets between routers. As will be apparent to one of skill in the
10 art, the number of pairs of routers that need to be monitored in such a scenario grows as a quadratic of N , where N is the number of network routers making up the network. Thus, such a measurement technique involves measurements on the order of N^2 ($O(N^2)$).

Once generated, the measurement data is collected and processed. The
15 measurement data can then be made available to other applications, such as for the tuning of network parameters and the like. As can be seen from the complexity of the technique ($O(N^2)$), this measurement scheme does not scale well in large networks as the number of specified pairs of routers to be monitored increases dramatically. In such cases, the resulting network traffic due to probe packets can be large and,
20 therefore, unsustainable as a result of the bandwidth consumed thereby.

This problem of scalability may be further compounded by the fact that networks being deployed are diff-serv (DS) enabled. In such cases, delay and jitter characteristics must be monitored for every DS-codepoint in use in the network. Diff-serv enabled networks offer a range of data transfer services that are differentiated on
25 the basis of performance experienced by packets belonging to a particular aggregated set of applications or flows. An application requests a specific level of performance on a packet-by-packet basis, by marking the type-of-service (ToS) field in each IP packet with a specific value, also called DS-codepoint. This value effectively

specifies how an enterprise network or a service provider network processes and forwards each packet along each hop in the network.

Moreover, it is often important which routers and links are involved in the measurements that are made. For example, it is usually preferable to make the
5 necessary measurements from a smaller number of routers. This simplifies the administration of the network measurement tools because there are fewer installations, and fewer instances of the tools to run. Additionally, it is often desirable to avoid sending measurement traffic over already-congested links. Preferably, such traffic is sent over links having at least a modicum of excess bandwidth, so as not to interfere
10 with actual network traffic. Meeting these and other such objectives simplifies network administration by minimizing the need for user interaction and a network's administrative burden.

What is therefore needed is a method and apparatus for the measurement of delays encountered by network traffic in traversing a network, the complexity of
15 which preferably grows at a rate less than $O(N^2)$. More preferably, the complexity of the measurement scheme should grow at a rate that increases at most linearly with the number of network routers (designated herein as N) and the number of network links (designated herein as M). Moreover, such a technique should address the situation in which an operator desires some modicum of control over the nodes and links over
20 which measurements are made.

Summary of the Invention

The present invention, in various embodiments, addresses the aforementioned problems by providing techniques for measuring network performance metrics, such as delay encountered between nodes in a network, and delay-jitter (jitter) observed
25 therein. Advantageously, embodiments of the present invention allow such measurements to be made between preferred nodes of the network, without generating excessive network traffic, and are capable of accounting for delays both between nodes (the delay over inter-node links) and within nodes. Also advantageously,

embodiments of the present invention are less demanding from a measurement perspective, being on the order of $N+M$ ($O(N+M)$).

A delay measurement technique according to an embodiment of the present invention is based on the precept, ascertained by the inventors, that a link between
5 network nodes will often contribute to the delay encountered between several different pairs of network nodes. Such a technique identifies the path between each pair of nodes by a list of links that form the path. Paths that are orthogonal are treated as being necessary for determining the delays encountered between nodes. Once delay and jitter over the requisite set of orthogonal paths have been measured, delay over all
10 other paths can be determined in terms of the delay and jitter encountered over one or more of these orthogonal paths. Such a technique also lends itself to a vector/matrix representation of delay and jitter over the paths, and the use of matrix manipulation techniques in deriving delay and jitter.

The techniques described herein provide several advantages. Such techniques
15 permit a network operator some measure of control over the network elements and network links involved in the measurements taken by such a procedure, allowing the operator to minimize the number of network elements involved, traffic over congested links and so on.. These techniques also address the fundamental concern of a measurement scheme's scalability in monitoring delay and jitter in increasingly large
20 networks, with known or partly known topology and routing. The techniques result in a significant reduction in traffic overhead due to exchange of probe packets when compared to existing techniques. The savings can be especially significant in networks having a large requirements set. Further, in a network employing TCP/IP protocols, the methods described herein separately accounts for IP-level transfer delay
25 over a link and the delay in processing probe packets within a router (i.e., the delay experienced in generating and/or receiving such packets), thereby helping to isolate and identify links that are over or under provisioned. Simplicity and ease of implementation are also attractive features of a technique according to the present invention.

In one embodiment of the present invention, a method of determining a network performance metric in a network is described. The network includes a number of network elements and a number of links. Each of the network elements is coupled to at least one other of the network elements by at least one of the links. The method includes forming a first set of network element pairs, ordering a first number of network element pairs, forming a second set of network element pairs, measuring a measured network performance metric between a first network element pair and computing a computed network performance metric. The first set of network element pairs include a number of pairs of the network elements. The ordering of a first number of network element pairs includes network element pairs in the first set of network element pairs. The second set of network element pairs includes network element pairs in the first set of network element pairs. The measurement is taken between a first network element pair. The first network element pair includes a first network element and a second network element of one of the network element pairs in the second set of network element pairs. The computed network performance metric is computed between a second network element pair using the measured network performance metric. The second network element pair includes a first network element and a second network element of the network element pair in the first set of network element pairs.

The foregoing is a summary and thus contains, by necessity, simplifications, generalizations and omissions of detail; consequently, those skilled in the art will appreciate that the summary is illustrative only and is not intended to be in any way limiting. As will also be apparent to one of skill in the art, the operations disclosed herein may be implemented in a number of ways, and such changes and modifications may be made without departing from this invention and its broader aspects. Other aspects, inventive features, and advantages of the present invention, as defined solely by the claims, will become apparent in the non-limiting detailed description set forth below.

BRIEF DESCRIPTION OF THE DRAWINGS

The present invention may be better understood, and its numerous objects, features, and advantages made apparent to those skilled in the art by referencing the accompanying drawings.

5 Fig. 1A is a block diagram illustrating a computer system.

Fig. 1B is a block diagram illustrating the various sources of delay encountered by network traffic transiting a network.

Fig. 2 is a block diagram depicting a network in which a delay measurement technique according to an embodiment of the present invention can be practiced.

10 Fig. 3 is a flow diagram illustrating a process for computing the round-trip delay between pairs of routers according to the present invention.

Fig. 4 is a flow diagram illustrating a scheme according to the present invention for ordering router pairs in a requirements set.

15 Fig. 5 is a flow diagram illustrating a method according to the present invention for identifying a preferred measurements set.

Fig. 6 is a block diagram illustrating an exemplary enterprise network in which a method according to an embodiment of the present invention can be practiced.

The use of the same reference symbols in different drawings indicates similar or identical items.

20 **DETAILED DESCRIPTION OF THE INVENTION**

The following is intended to provide a detailed description of an example of the invention and should not be taken to be limiting of the invention itself. Rather, any number of variations may fall within the scope of the invention which is defined in the claims following the description.

Introduction

Embodiments of the measurement techniques described herein address the central issue of scalability. These methods allow (a) network performance metrics, such as delay and jitter, to be measured between a significantly smaller subset of pairs of routers, and (b) the corresponding metrics for the remaining specified pairs of routers to be computed based on these measurements. These methods include (a) the identification of network paths for which measurements must be undertaken, and (b) the computation of the metrics for the remaining specified pairs of routers for which measurements are not taken. Advantageously, the present invention allows control over which routers and links such measurements are taken.

A measurement scheme according to the present invention employs a measurements set and a requirements set, which are defined herein as follows:

1. The "requirements set" is defined herein as the specified collection of pairs of routers for which delay and jitter metrics are to be obtained.
2. The "measurements set" is defined herein as the set of router pairs for which delay/jitter measurements are to be taken. A fundamental property of the measurements set is that, based on measurements thus obtained, one can accurately compute (estimate) delay/jitter metrics for router pairs specified in the requirements set.

The methods described herein are based on a technique that divides the end-router-to-end-router path in terms of processing delay within the source and destination routers and the IP-level transfer delay over intermediate link(s) (and router(s), if any) in the network path between the pair of end-routers. Moreover, techniques according to embodiments of the present invention can also be used in scenarios in which the internal topology of the network being analyzed is partly unknown. Central to these concepts is (a) the identification of a minimal measurements set from a given requirements set and (b) the estimation of delay and jitter for pairs of routers in the requirements set. It will be noted that, although the

techniques disclosed herein are generally applicable to various network protocols and topologies, these techniques are discussed herein in terms of TCP/IP protocols.

Depending upon the order in which the pairs of routers in the requirements set (Φ) are listed and processed, one of many possible measurements sets can result. The set of router pairs between which measurements should be made is thus dependent (at least in part) on the ordering of the requirements set. The specific measurements set (Ω) sought is one that requires measurements to be made between one or more preferred set of router pairs. For example, in a given network, a requirements set will contain a given assemblage of router pairs. For the given requirements set, there will be a set of router pairs that, once delay and delay-jitter are measured between those routers, allow delays between other router pairs to be computed (estimated). These router pairs make up the measurements set.

However, the ordering of router pairs in the requirements set affects the determination of pairs of routers in the measurements set. Typically, the router pairs in the requirements set can be ordered in any number of ways. However, it is often desirable to control which router pairs are in the measurements set. The inventors discovered that the router pairs in the measurements set can be controlled by altering the ordering of router pairs in the requirements set. This ability to control which router pairs are in the measurements set is desirable because such control can allow an operator to minimize the number of routers on which active network monitoring tools (ANMTs) must be run (or links over which measurement traffic must travel), to minimize measurement traffic over critical links and to avoid the need to run ANMTs on critical or over-stressed routers, among numerous other advantages.

To provide such advantages, a measurement scheme according to the present invention identifies the measurements set from a given requirements set by:

1. Specifying the sequence of routers, in decreasing priority, from which
5 measurements are initiated.
2. Expressing delay between each pair of routers in the requirements set, Φ , as a linear function of delay components and constructing the matrix F .
3. Using Gauss elimination (or another equivalent thereof) to decompose matrix
10 F into a lower-triangular matrix A and an upper-triangular matrix F' , till all independent row vectors of F have been identified.
4. Using the non-zero row vectors in F' to determine the measurements set, Ω .

The matrix F' identifies the independent row vectors, while matrix A allows a dependent row of F to be expressed in terms of the independent rows of F . Pairs of routers that correspond to the independent rows of F directly help one to determine the
15 measurements set, Ω .

Other than resulting in considerably reduced network traffic due to probe packets, a monitoring scheme based on the present invention enables one to compute (estimate) the IP-level transfer delay over each router-to-router link in the network. Such an estimate factors out the delay in processing (generating/receiving) probe
20 packets within a router. This is desirable because the delay in processing probe packets can be significant as a result of probes being generated and received within the application layer. It will be apparent to one skilled in the art that an assessment of IP-level transfer delay over each link assists a network administrator in identifying congested links in a functioning network.

25 The methods described above are applicable to networks for which (a) topology and routing information is available, and (b) changes in topology and routing are infrequent. These methods may also be extended to apply to networks that are built around a network backbone, the topology and routing of which are unknown.

Such a method directly provides a measurements set and a scheme for estimating delay and jitter between the routers on such a network. As a result, one is also able to apportion the delay-components between the enterprise network and the backbone network.

5 The techniques described herein provide several advantages. Chief among them is the ability to control, to the extent feasible, which routers and links are involved in actual measurements. Moreover, these techniques address the fundamental concern of a measurement scheme's scalability in monitoring delay and jitter in increasingly large networks, with known or partly known topology and
10 routing. The techniques result in a significant reduction in network traffic overhead due to exchange of probe packets when compared to existing techniques. The savings can be especially significant in networks having a large requirements set. Further, the methods described herein separately account for IP-level transfer delay over a link and the delay in processing probe packets within a router (i.e., the delay experienced in
15 generating and/or receiving such packets), thereby helping to isolate and identify links that are over or under provisioned. Simplicity and ease of implementation are also attractive feature of these techniques.

Measurements Using a Preferred Measurements Set

20 An active network monitoring tool (referred to generically herein as an ANMT) can be used to measure round-trip delay, delay variation (jitter) and packet loss between a specified set of pairs of routers (or a network switch with a router switch module or an attached one-arm router) in a large diff-serv enabled enterprise network. By design, ANMT software running as an application on a router (e.g., a source router), sends a sequence of user datagram protocol (UDP) packets (also
25 known as probe packets) to a specified port on a remote device (e.g., a destination router). Based on information contained in UDP responses received from the destination, the ANMT software running on the source router computes various statistics, such as average round-trip delay, standard deviation of delay jitter, and average fraction of packets dropped. An ANMT may also be used to test TCP-level

connectivity or to estimate one-way packet loss and delay-variation. Additionally, an ANMT allows monitoring of performance between devices in a network which offers differentiated services.

A method according to one embodiment of the present invention permits the use of a network measurement tool (e.g., an ANMT) to measure round-trip delay and jitter between a much smaller subset of pairs of routers (represented herein by the notation Ω), in order to estimate delay and jitter between pairs of routers in a specified set (represented herein by the notation Φ). This subset Ω includes at most $(N+M)$ pairs of routers, where N is the number of enterprise routers, and M is the number of links. This is a significant improvement over the complexity of measuring delay between every pair of routers in the set Φ using a tool such as an ANMT. It will be noted that the number of pairs of routers in Φ may be as large as $N(N+1)/2$, or $O(N^2)$. Second, the techniques discussed herein recognize that delay encountered in generating, receiving and processing probe packets within the transport and application layers in a router may be significant.

Fig. 1A depicts a block diagram of a computer system 10 suitable for implementing the present invention. Computer system 10 includes a bus 12 which interconnects major subsystems of computer system 10 such as a central processor 14, a system memory 16 (typically RAM, but which may also include ROM, flash RAM, or the like), an input/output controller 18, an external audio device such as a speaker system 20 via an audio output interface 22, an external device such as a display screen 24 via display adapter 26, serial ports 28 and 30, a keyboard 32 (interfaced with a keyboard controller 33), a storage interface 34, a floppy disk drive 36 operative to receive a floppy disk 38, and a CD-ROM drive 40 operative to receive a CD-ROM 42. Also included are a mouse 46 (or other point-and-click device, coupled to bus 12 via serial port 28), a modem 47 (coupled to bus 12 via serial port 30) and a network interface 48 (coupled directly to bus 12).

Bus 12 allows data communication between central processor 14 and system memory 16, which may include both read only memory (ROM) or flash memory

(neither shown), and random access memory (RAM) (not shown), as previously noted. The RAM is generally the main memory into which the operating system and application programs are loaded and typically affords at least 16 megabytes of memory space. The ROM or flash memory may contain, among other code, the Basic Input-Output system (BIOS) which controls basic hardware operation such as the interaction with peripheral components. Applications resident with computer system 10 are generally stored on and accessed via a computer readable medium, such as a hard disk drive (e.g., fixed disk 44), an optical drive (e.g., CD-ROM drive 40), floppy disk unit 36 or other storage medium. Additionally, applications may be in the form of electronic signals modulated in accordance with the application and data communication technology when accessed via network modem 47 or interface 48.

Storage interface 34, as with the other storage interfaces of computer system 10, may connect to a standard computer readable medium for storage and/or retrieval of information, such as a fixed disk drive 44. Fixed disk drive 44 may be a part of computer system 10 or may be separate and accessed through other interface systems. Many other devices can be connected such as a mouse 46 connected to bus 12 via serial port 28, a modem 47 connected to bus 12 via serial port 30 and a network interface 48 connected directly to bus 12. Modem 47 may provide a direct connection to a remote server via a telephone link or to the Internet via an internet service provider (ISP). Network interface 48 may provide a direct connection to a remote server via a direct network link to the Internet via a POP (point of presence).

Many other devices or subsystems (not shown) may be connected in a similar manner. Conversely, it is not necessary for all of the devices shown in Fig. 1A to be present to practice the present invention. The devices and subsystems may be interconnected in different ways from that shown in Fig. 1A. The operation of a computer system such as that shown in Fig. 1A is readily known in the art and is not discussed in detail in this application. Code to implement the present invention may be operably disposed or stored in computer-readable storage media such as one or more of system memory 16, fixed disk 44, CD-ROM 42, or floppy disk 38.

It will be noted that the variable identifier "N" is used in several instances in Fig. 1A to more simply designate the final element of a series of related or similar elements. The repeated use of such variable identifiers is not meant to imply a correlation between the sizes of such series of elements, although such correlation may exist. The use of such variable identifiers does not require that each series of elements has the same number of elements as another series delimited by the same variable identifier. Rather, in each instance of use, the variable identified by "N" may hold the same or a different value than other instances of the same variable identifier.

Moreover, regarding the signals described herein, those skilled in the art will recognize that a signal may be directly transmitted from a first block to a second block, or a signal may be modified (e.g., amplified, attenuated, delayed, latched, buffered, inverted, filtered or otherwise modified) between the blocks. Although the signals of the above described embodiment are characterized as transmitted from one block to the next, other embodiments of the present invention may include modified signals in place of such directly transmitted signals as long as the informational and/or functional aspect of the signal is transmitted between blocks. To some extent, a signal input at a second block may be conceptualized as a second signal derived from a first signal output from a first block due to physical limitations of the circuitry involved (e.g., there will inevitably be some attenuation and delay). Therefore, as used herein, a second signal derived from a first signal includes the first signal or any modifications to the first signal, whether due to circuit limitations or due to passage through other circuit elements which do not change the informational and/or final functional aspect of the first signal.

The foregoing described embodiment wherein the different components are contained within different other components (e.g., the various elements shown as components of computer system 10). It is to be understood that such depicted architectures are merely exemplary, and that in fact many other architectures can be implemented which achieve the same functionality. In an abstract, but still definite sense, any arrangement of components to achieve the same functionality is effectively "associated" such that the desired functionality is achieved. Hence, any two

components herein combined to achieve a particular functionality can be seen as "associated with" each other such that the desired functionality is achieved, irrespective of architectures or intermedial components. Likewise, any two components so associated can also be viewed as being "operably connected", or

5 "operably coupled", to each other to achieve the desired functionality.

Fig. 1B depicts a block diagram illustrating the various sources of delay encountered by network traffic transiting a network 100. While a method according to an embodiment of the present invention may be practiced using any one of a number of protocols, Fig. 1B is illustrated in terms of a TCP/IP network protocol.

10 Delay can be apportioned into monitor application response delay (exemplified in Fig. 1B by monitor application response delays 105 and 110) and network layer transfer delays (exemplified in Fig. 1B by network layer transfer delays 115(1)-(N)) over an (end-to-end) network path 120 between a source router 125 and a destination router 130. In transiting network 100, network path 120 may also transit one or more

15 intermediate routers (exemplified in Fig. 1B by an intermediate router 135). Source router 125, intermediate router 135 and destination router 130 are also designated as R_1 , R_2 , and R_N , indicating the progression of network traffic transiting network 100. In a TCP/IP environment such as that of network 100, monitor application delays 105 and 110 include, respectively, delays attributable to a network monitor 140, and

20 transport layer 145, a transport layer 146 and a network monitor 141. In a similar fashion, network layer transfer delays 115(1)-(N) include delays attributable to moving IP packets through and between IP layers 150, 151 and 152, intermediate layers 155, 156 and 157, and physical layers 160, 161 and 162, in the manner shown in Fig. 1B. As can be seen in Fig. 1B, the two principle types of delay components

25 contributing to the round-trip delay between a pair of routers such as source router 125 and destination router 130 when measured using an ANMT can be defined as:

$d_j, j = 1, 2, \dots, M$, the round-trip IP-level delay in transferring an IP packet over a link, L_j , between two adjacent routers; and

5 $s_i, i = 1, 2, \dots, N$, the time to generate, receive and process a probe packet within transport and application layers in a router, R_i .

The symbols used herein in describing techniques according to various embodiments of the present invention for estimating round-trip delay and jitter between a specified set of pairs of routers are now presented.

10 Fig. 2 is a block diagram depicting a network 200 that illustrates the components of the delay existing between various pairs of routers that make up network 200. As is illustrated in Fig. 2, network 200 includes a number of routers (exemplified in Fig. 2 by routers 210(1)-(6)) which are coupled to one another by a number of interconnecting links (exemplified in Fig. 2 by links 220(1)-(9)). In order
15 to facilitate the following discussions, routers 210(1)-(6) are also designated R_1 - R_6 , respectively. In similar fashion, links 220(1)-(9) are referred to using L_1 - L_9 , respectively. In the general sense, network 200 can be described as including N routers, $R_i, i = 1, 2, \dots, N$, and M interconnecting links, $L_j, j = 1, 2, \dots, M$. A route between a given pair of routers, $\pi_k = (R_i, R_j)$, is specified as an ordered list of
20 intermediate links that network traffic between R_i and R_j traverses:

$$\text{link_path}(R_i, R_j) = \langle L_p, L_q, \dots, L_r \rangle \quad (1)$$

25 That is, network traffic moving from source router R_i to destination router R_j traverse the links, L_p, L_q, \dots , and L_r in that order, and *vice-versa*, in the other direction.

For a given DS codepoint, the routes between routers, R_i and R_j , of network 200 are given in Table 1. It will be noted that the routes in Table 1 are specified for a given DS codepoint, and in terms of intermediate links.

	R_1	R_2	R_3	R_4	R_5	R_6
R_1	-	L_1	L_1, L_2	L_3	L_3, L_8	L_3, L_8, L_9
R_2		-	L_2	L_4	L_5	L_2, L_7
R_3			-	L_6, L_8	L_6	L_7
R_4				-	L_8	L_8, L_9
R_5					-	L_9
R_6						-

Table 1. Routing table for network given in Fig. 2.

As noted, the two principal types of delay components that contribute to the round-trip delay between a pair of routers when measured using an ANMT can be defined as d_j , $j = 1, 2, \dots, M$ (the round-trip IP-level delay in transferring an IP packet over a link, L_j), and s_i , $i = 1, 2, \dots, N$ (the time to generate, receive and process a probe packet within transport and application layers in a router, R_i).

As a result, the round-trip delay between any given pair of routers, R_i and R_j , is given by:

$$10 \quad \text{Delay}(R_i, R_j) = s_i + d_p + d_q + \dots + d_r + s_j \quad (2)$$

assuming that the route between R_i and R_j is known and specified in terms of the intermediate links. That is,

$$15 \quad \text{link_path}(R_i, R_j) = \langle L_p, L_q, \dots, L_r \rangle \quad (3)$$

When used between routers, R_i and R_j , a delay monitoring tool such as an ANMT measures $\text{Delay}(R_i, R_j)$. Such measurements may include measurement errors.

The requirements set and measurements set can now be defined using the framework set forth above.

The user-specified requirements set of P number of pairs of routers between which round-trip delay is to be measured is given by

$$25 \quad \Phi = \{\pi_k, k = 1, 2, \dots, P\} \quad (4)$$

where

$$\pi_k = (R_i, R_j) \quad (5)$$

and so

$$\Phi = \{(R_i, R_j), i, j = 1, 2, \dots, P\} \quad (6)$$

Normally, the subset will only be a small fraction of all possible pairs of routers. As noted, because round-trip delay or jitter is monitored, it is immaterial whether the pair of routers is specified as $\pi_k = (R_i, R_j)$ or as $\pi_k = (R_j, R_i)$.

The estimate is based on measurements between a subset of pairs of routers (and indeed what can be a minimal subset of router pairs), which may be referred to by Ω . That is, for every $(R_i, R_j) \in \Phi$, the delay between the pair of routers (R_i, R_j) can be computed (or estimated) in terms of measured delay between one or more pairs of routers $\{(R_p, R_q)\} \in \Omega$. The set, Ω , is also referred to herein as the measurements set. A method according to the present invention identifies the measurements set, Ω , for a given requirements set, Φ .

Determination of the Measurements Set

A method according to an embodiment of the present invention is now described that is capable of identifying the collection of pairs of routers, or Ω , such that the delay between a given pair of routers $(R_i, R_j) \in \Phi$ can be computed from measurements between a subset of pairs of routers $(R_p, R_q) \in \Omega$.

If the number of pairs of routers in Φ is defined as P (or, $P = |\Phi|$), the round-trip delay between the k^{th} pair of routers, $(R_i, R_j) \in \Phi$ is denoted by $z_k = \text{Delay}(R_i, R_j)$.

In view of the definition of delay components, it can be seen that, for every $k = 1, 2, \dots, P$:

$$z_k = \text{Delay}(R_i, R_j) = s_i + d_p + d_q + \dots + d_r + s_j \quad (7a)$$

assuming that the route between R_i and R_j is known and given by

$$\text{link_path}(R_i, R_j) = \langle L_p, L_q, \dots, L_r \rangle \quad (7b)$$

Using a vector notation, $z_k = \text{Delay}(R_i, R_j)$ may be re-written as

$$z_k = \text{Delay}(R_i, R_j) \quad (7c)$$

$$= s_i + d_p + d_q + \dots + d_r + s_j \quad (7d)$$

$$= F_k x \quad (7e)$$

10 where

$$F_k = [0 \dots 0 \ 1 \ 0 \dots 0 \ 1 \ 0 \dots 0 \ 1 \ 0 \dots 0 \ 1 \ 0 \dots 0 \ 1 \ 0 \dots 0] \quad (8)$$

15 is a row vector of size $N+M$, and the 1 's in F_k appear in columns p, q, \dots, r , and in columns $M+i$ and $M+j$; and

x is a vector of size $N+M$ (the delay-components vector):

$$x = \begin{bmatrix} d_1 \\ d_2 \\ \dots \\ d_M \\ s_1 \\ s_2 \\ \dots \\ s_N \end{bmatrix} \quad (9)$$

As noted, the round-trip delay between the specified pairs of routers in $\Phi = \{\pi_k = (R_i, R_j), k = 1, 2, \dots, P\}$, $z_k = F_k x$ and $k = 1, 2, \dots, P$, can be re-written as a vector equation

$$z = F x \quad (10a)$$

where

$$z = \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \quad (10b)$$

$$\begin{bmatrix} \dots \\ z_P \end{bmatrix}$$

and

5

$$F = \begin{bmatrix} F_1 \\ F_2 \\ \dots \\ F_P \end{bmatrix} \quad (10c)$$

10

The $P \times (N+M)$ matrix, F , plays an important role in determining the subset of pairs of routers between which delay measurements are necessary.

Consider the row vectors, F_k , $k = 1, 2, \dots, P$ and let Q be the maximum number of independent row vectors. As can be seen, Q is equal to the rank of matrix F (i.e., $\text{Rank}(F)$). Without loss of generality, let $F_1 F_2 \dots F_Q$ be the independent row vectors of matrix F . This is without loss of generality because the row vectors F_k in F (and similarly the corresponding pairs of routers in Φ) can be re-arranged, if necessary. Then, every row vector F_k , $k = Q+1, Q+2, \dots, P$, can be expressed as a linear combination of row vectors, F_k , $k = 1, 2, \dots, Q$.

20

The relationship between the linearly independent and linearly dependent rows of F is now examined. If F_1, F_2, \dots, F_Q is a maximal set of linearly independent rows of F , then row vectors $F_{Q+1}, F_{Q+2}, \dots, F_P$ can be expressed as a linear combination of F_1, F_2, \dots, F_Q . In other words, F_k can be expressed in terms of:

25

$$F_k = \sum_{i=1, \dots, Q} (\alpha_{k,i} F_i), k = Q+1, Q+2, \dots, P \quad (11)$$

The constants $\alpha_{k,i}$ may be re-organized in the form of a row vector (of size Q) in the following manner:

30

$$\alpha_k = [\alpha_{k,1}, \alpha_{k,2}, \dots, \alpha_{k,Q}], k = Q+1, Q+2, \dots, P \quad (12)$$

The values of α_k may be aggregated to form the $P \times Q$ matrix, A , which can be defined as:

$$A = \begin{bmatrix} I & \\ \alpha_{Q+1} & \\ \alpha_{Q+2} & \\ \dots & \\ \alpha_P & \end{bmatrix} \quad (13)$$

where the matrix, I , is a $Q \times Q$ matrix.

Because delay (z_k) is a random variable, delay-jitter between pair of routers ($\pi_k = (R_i, R_j)$) can, be defined as follows:

$$\text{Delay-Jitter}(R_i, R_j) = \sqrt{E\{ (z_k - E\{z_k\})^2 \}} \quad (14)$$

where $E\{.\}$ is the expectation operation.

A measurements set can be defined as the set of Q number of pairs of routers between which a network administrator uses an ANMT to measure round-trip delay, and is given by

$$\Omega = \{\pi'_k = (R_i, R_j), k = 1, 2, \dots, Q\} \quad (15)$$

Router-to-router delay and delay-jitter is measured for the measurements set using an ANMT. When used between a pair of routers, $\pi_k = (R_i, R_j)$, an ANMT measures Delay(R_i, R_j) to yield

$$y_k = z_k + v_k \quad (16a)$$

Using an ANMT, one typically makes several measurements, y_k^i , during the time interval ($t_0, t_0 + dt$), and returns the statistical average,

$$y_k = \text{statistical_avg}(y_k^i) \quad (16b)$$

as well as the observed standard deviation

$$\gamma_k = \sqrt{\text{statistical_avg}((y_k - y_k^i)^2)} \quad (16c)$$

- 5 The above delay and delay-jitter measurements may be organized in the form of a vector, y , of size Q :

$$y = \begin{bmatrix} y_1 \\ y_2 \\ \dots \\ y_Q \end{bmatrix} \quad (17)$$

and a matrix, Γ , of size $Q \times Q$:

$$\Gamma = \begin{bmatrix} \gamma_1 & 0 & 0 & \dots & 0 \\ 0 & \gamma_2 & 0 & \dots & 0 \\ \dots & & & & \\ 0 & 0 & \dots & 0 & \gamma_Q \end{bmatrix} \quad (18)$$

- 20 Estimates can then be made for router pairs in the requirements set, Φ . The delay estimate vector, Δ , is then calculated. More specifically, an estimate of delay between the pair of routers, $\pi_k = (R_i, R_j) \in \Phi$, is denoted by Δ_k . The vector, Δ , of size P , denotes an estimate of delay for all pairs of routers in Φ .

- 25 An estimate of delay-jitter between the pair of routers, $\pi_k = (R_i, R_j) \in \Phi$, is denoted by σ_k . The vector, σ , of size P , denotes an estimate of delay-jitter for all pairs of routers in Φ .

In regard to a measurement method according to certain embodiments of the present invention, certain assumptions concerning the network being analyzed are now summarized. The network is assumed to have the following characteristics:

- 30
- For the embodiments discussed above, it is assumed that the network's topology is known. It will be noted that, because a route between a pair of routers is specified

in terms of links, there is no constraint on the existence of parallel links between a pair of enterprise routers.

- Further, it is preferable that the topology not change frequently. If the topology of the network changes, then the procedure described herein should be carried out afresh.
- For each DS codepoint, the route between every pair of enterprise routers should be known.
- The route between every pair of enterprise routers for a given DS codepoint is symmetric.
- The delay and jitter characteristics between a pair of routers may be different in the two directions.

In other words, certain of the procedures described herein are preferably used together with techniques that monitor changes in topology and in routing. Once a change in topology or in routing is detected by these techniques, the method according to an embodiment of the present invention should be performed once again.

As a consequence of the foregoing manipulations, certain results can be deduced. As outlined below, these results include:

1. the fact that, for every $(R_i, R_j) \in \Phi$, $\text{Delay}(R_i, R_j)$ can be expressed as a linear combination of one or more $\text{Delay}(R_p, R_q)$, $(R_p, R_q) \in \Omega$; and
2. the fact that the number of router-to-router delay measurements required to estimate delay between an arbitrarily specified subset of pairs of routers is at most $M+N$.

These results are set forth in detail below.

First Result

The first result is determined as follows. Consider the matrix $F = [F_1 F_2 \dots F_p]^T$, corresponding to the given set of router pairs in the requirements set, Φ . Then, the minimal subset of pairs of routers, Ω , between which delay must be measured is given by the collection of all pairs of routers, (R_p, R_q) , that correspond to the maximal set of independent rows of F , with respect to $\{F_k, k = 1, 2, \dots, Q\}$. In other words, the measurements set can be represented as:

$$\Omega = \{(R_p, R_q), \text{ such that } \text{Delay}(R_p, R_q) = F_k x, k = 1, 2, \dots, Q\} \quad (19)$$

As a consequence, for every $(R_i, R_j) \in \Phi$, $\text{Delay}(R_i, R_j)$ can be expressed as a linear combination of one or more $\text{Delay}(R_p, R_q)$, where $(R_p, R_q) \in \Omega$.

The above result is illustrated by the following example. Consider the network of Fig. 2, where $N=6$, $M=9$. If the requirements set, $\Phi = \{(R_3, R_4) (R_3, R_5) (R_4, R_6) (R_5, R_6)\}$, then from routing table given in Table 1 for a given DS codepoint:

$$z_1 = \text{Delay}(R_3, R_4) = s_3 + d_6 + d_8 + s_4 \quad (20a)$$

$$z_2 = \text{Delay}(R_3, R_5) = s_3 + d_6 + s_5 \quad (20b)$$

$$z_3 = \text{Delay}(R_4, R_6) = s_4 + d_8 + d_9 + s_6 \quad (20c)$$

$$z_4 = \text{Delay}(R_5, R_6) = s_5 + d_9 + s_6 \quad (20d)$$

That is,

$$z = [z_1 z_2 z_3 z_4]^T = F x \quad (20e)$$

where

$$x = [d_1, d_2, \dots, d_9, s_1, s_2, \dots, s_6]^T \quad (21)$$

The F matrix is a 4x15 matrix of the form:

	D ₁	D ₂	D ₃	D ₄	D ₅	D ₆	D ₇	D ₈	D ₉	S ₁	S ₂	S ₃	S ₄	S ₅	S ₆
z ₁						1		1				1	1		
z ₂						1						1		1	
z ₃								1	1				1		1
z ₄									1					1	1

It is verified subsequently herein that Rank(F) = 3, and that z₂, z₃ and z₄ are

- 5 independent row vectors, and that z₁ = z₂ + z₃ - z₄. Thus, the measurements set, $\Omega = \{(R_3, R_5), (R_4, R_6), (R_5, R_6)\}$.

Using a different requirements set for network 200, if

$$10 \quad \Phi = \{(R_1, R_2), (R_2, R_3), (R_1, R_4), (R_2, R_4), (R_2, R_5), (R_3, R_5), (R_3, R_6), (R_4, R_5), (R_5, R_6), (R_1, R_3), (R_2, R_6), (R_1, R_5), (R_4, R_6), (R_1, R_6)\} \quad (22)$$

the resulting matrix F is

	D ₁	D ₂	D ₃	D ₄	D ₅	D ₆	D ₇	D ₈	D ₉	S ₁	S ₂	S ₃	S ₄	S ₅	S ₆
z ₁	1									1	1				
z ₂		1									1	1			
z ₃			1							1			1		
z ₄				1							1		1		
z ₅					1						1			1	
z ₆						1						1		1	
z ₇							1					1			1
z ₈								1					1	1	
z ₉									1					1	1
z ₁₀	1	1								1		1			
z ₁₁		1					1				1				1
z ₁₂			1					1		1				1	
z ₁₃								1	1				1		1
z ₁₄			1					1	1	1					1

15 It can be shown that the matrix F has rank 13, and that the first 13 rows of F are independent. Further, z₁₄ = z₁₂ + z₁₃ - z₈. Thus $\Omega = \{(R_1, R_2), (R_2, R_3), (R_1, R_4), (R_2,$

$R_4), (R_2, R_5), (R_3, R_5), (R_3, R_6), (R_4, R_5), (R_5, R_6), (R_1, R_3), (R_2, R_6), (R_1, R_5), (R_4, R_6)\}$.

It will be noted that Ω is also the subset of measurements required to estimate the delay between an arbitrary pair of routers. The second result is now examined.

Second Result

5 The minimal subset of pairs of routers, between which delay should be measured, so that delay between every other pair of routers can be obtained, can also be determined. The number of router-to-router delay measurements required to estimate delay between an arbitrarily specified subset of pairs of routers is at most $M+N$. In other words:

$$10 \quad Q = |\Omega| = \text{Rank}(F) \leq \min(|\Phi|, M+N) \quad (23)$$

This is so, because the matrix, F , is a $|\Phi| \times (M+N)$, the rank of which, with respect to Q , is at most $|\Phi|$ or $(M+N)$, whichever is smaller. The theoretical underpinnings of
15 the foregoing results are now examined.

Theoretical Underpinnings Of The Measurement Scheme

A measurement scheme according to certain embodiments of the present invention, as described herein, makes Q measurements between the collection of Q pairs of routers, specified by Ω . It will be noted that $Q=|\Omega|$, and that Ω is the
20 collection of pairs of routers, (R_p, R_q) , that correspond to the maximal set of independent rows of F , with respect to $\{F_k, k = 1, 2, \dots, Q\}$ (i.e., the measurements set).

Network analysis agents (e.g., ANMTs) can be executed, for example, as transport applications in two routers, R_i and R_j , to measure round-trip delay and jitter
25 between R_i and R_j . Such a measurement, denoted y_k yields:

$$y_k = \text{Delay}(R_i, R_j) + v_k = s_i + d_p + d_q + \dots + d_r + s_j + v_k \quad (24)$$

It is assumed that the intermediate links between the two routers are L_p, L_q, \dots, L_r , that the measurement error is v_k . As noted, processing delays encountered by probe packets within source and destination routers, R_i and R_j , respectively, are defined as s_i and s_j , and d_p, d_q, \dots, d_r are round-trip IP-layer transfer delays encountered by probe packets over intermediate links along the route

$$\text{link_route}(R_i, R_j) = \langle L_p, L_q, \dots, L_r \rangle \quad (25)$$

The k^{th} measurement, y_k , $k = 1, 2, \dots, Q$, of round-trip delay between the k^{th} pair of routers, R_i and R_j , in Ω , may be re-written as:

$$y_k = \text{Delay}(R_i, R_j) + v_k = s_i + d_p + d_q + \dots + d_r + s_j + v_k \quad (26)$$

$$= F_k x + v_k$$

where the row vector (of size $M+N$)

$$F_k = [0 \dots 0 \ 1 \ 0 \dots 0 \ 1 \ 0 \dots 0 \ 1 \ 0 \dots 0 \ 1 \ 0 \dots 0 \ 1 \ 0 \dots 0] \quad (27)$$

and the 1's in F_k appear in columns p, q, \dots, r , and in columns $M+i$ and $M+j$. The following can be defined:

$$y = [y_1 \ y_2 \ \dots \ y_Q]^T \quad (28a)$$

$$x = [d_1, d_2, \dots, d_M, s_1, s_2, \dots, s_N]^T \quad (28b)$$

$$v = [v_1 \ v_2 \ \dots \ v_Q]^T \quad (28c)$$

where vector y , of size Q , represents the collection of measurements, vector x , of size $M+N$, is the collection of delay components and vector v , of size Q , is the measurement error in y . The measurements, y_k , $k = 1, 2, \dots, Q$, taken together, may now be re-written in the form of a vector equation

$$y = H x + v \quad (29)$$

Consider, for example, the network given in Fig. 2, where $N=6$ and $M=9$. The routes between pairs of routers, R_i and R_j , are given in Table 1. Let $\Omega = \{(R_3, R_5) (R_4$
5 $R_6) (R_5, R_6)\}$ be the specified subset of pairs between which delay is measured. From Table 1,

$$10 \quad y_2 = \text{Delay}(R_4, R_6) + v_2 = s_4 + d_8 + d_9 + s_6 + v_2 \quad (30b)$$

$$y_3 = \text{Delay}(R_5, R_6) + v_3 = s_5 + d_9 + s_6 + v_3 \quad (30c)$$

$$15 \quad \mathbf{y} = \mathbf{H} \mathbf{x} + \mathbf{v} \quad (31)$$
$$\mathbf{y} = [\mathbf{y}_1 \ \mathbf{y}_2 \ \mathbf{y}_3]^T \quad (32a)$$

$$\mathbf{x} = [d_1, d_2, \dots, d_9, s_1, s_2, \dots, s_6]^T \quad (32b)$$

$$\mathbf{v} = [\mathbf{v}_1 \ \mathbf{v}_2 \ \mathbf{v}_3]^T \quad (32c)$$

	D ₁	D ₂	D ₃	D ₄	D ₅	D ₆	D ₇	D ₈	D ₉	S ₁	S ₂	S ₃	S ₄	S ₅	S ₆
Y ₁						1						1		1	
Y ₂								1	1				1		1
Y ₃									1					1	1

In this section, a methodology for computing the round-trip delay between every pair of routers within a given requirements set, $(R_i, R_j) \in \Phi$, based on

measurements between pairs of routers identified by the measurements set, Ω (a subset of the requirements set), is described. This approach is particularly useful when measurement errors are negligible.

As previously noted, the delay vector (z) corresponding to the requirements set
5 is given by

$$z = F x \quad (33)$$

while the measurement vector (y) corresponding to the measurements set is given by
10

$$y = H x \quad (34)$$

where

$$15 \quad H = [F_1^T F_2^T \dots F_Q^T]^T \quad (35)$$

Because $\{F_1, F_2, \dots, F_Q\}$ is the maximal set of independent rows of F , every row vector F_k can be expressed as a linear combination of F_1, F_2, \dots, F_Q .

As a result

$$20 \quad z = F x = A H x \quad (36a)$$

where

$$25 \quad A = [I \alpha^T]^T \quad (36b)$$

and the row vectors α_k , $k = Q+1, Q+2, \dots, P$ are used to express F_k in terms of F_1, F_2, \dots, F_Q (as discussed above with regard to Equation (13)).

30 The following provides a technique for allowing the network's operator to control (at least to some extent) which routers and links are involved in the measurements thus taken.

A Preferred Measurements Set

Depending upon the order in which the pairs of routers in the requirements set (Φ) are listed and processed, one of many possible measurements sets can result. The set of router pairs between which measurements are made is dependent (at least in part) on the ordering of the requirements set. The specific measurements set sought is the one that requires measurements to be made between one or more preferred set of router pairs. For example, in a given network, and let

$$\Phi = \{(R_1, R_2), (R_1, R_3), (R_1, R_4), (R_1, R_5), (R_1, R_6), (R_1, R_7), (R_2, R_3), (R_2, R_4), (R_2, R_5), (R_2, R_6), (R_2, R_7), (R_3, R_4), (R_3, R_5), (R_3, R_6), (R_3, R_7)\}$$

Given the option, an operator of such a network would prefer to use the measurement set:

$$\Omega^1 = \{(R_1, R_2), (R_1, R_3), (R_1, R_4), (R_1, R_5), (R_1, R_6), (R_1, R_7), (R_4, R_2)\}$$

instead of

$$\Omega^2 = \{(R_1, R_2), (R_1, R_3), (R_1, R_5), (R_1, R_6), (R_1, R_7), (R_2, R_7), (R_3, R_4)\}$$

because the former set, Ω^1 , requires that measurements be initiated from R_1 and R_2 only, while the latter, Ω^2 , requires that measurements be initiated from R_1 , R_2 , and R_3 . The former measurements set is thus referred to herein as a preferred measurements set, although such a measurements set can be referred to as a measurements set for the sake of simplicity.

A General Procedure For Employing An Ordered Measurements Set In Determining Delay For A Requirements Set

A procedure to deterministically compute the round-trip delay between pairs of routers, given by Φ , from measured round-trip delay between the Q number of ordered pairs of routers, identified by Ω , can be defined using the preceding results.

As can be seen, the computed round-trip delay, Δ , between the P pairs of routers in Φ is given by

$$\Delta = A y \quad (37)$$

where y_k , $k = 1, \dots, Q$, are delay measurements between the Q pairs of routers in Ω , and Δ_j , $j = 1, 2, \dots, P$, are delay between the P pairs of routers in Φ .

Fig. 3 illustrates a process of deterministically computing the round-trip delay between preferred pairs of routers. The process begins with the specification of the requirement set (step 300). Next, the router pairs in the requirements set are ordered in order to effect the operator's desires as to the ordering of the preferred measurements set (step 310). Delay components are then determined, and the delay components vector (x) constructed (step 320). Next, the F matrix is determined from the given requirements set and routing table (step 330). Once the F and x matrices have been constructed, the preferred measurements set, the F matrix's rank (Q) and the A matrix are calculated by performing the procedure preferred_measurements_set (step 340).

Preferred_measurements_set performs functions such as determining the maximal set of independent rows of F, viz. F_1, F_2, \dots, F_Q , in part by determining the rank of F (denoted herein by Q); constructing the A matrix by determining $\alpha_k = [\alpha_{k,1}, \alpha_{k,2}, \dots, \alpha_{k,Q}]$, such that $F_k = \sum_{i=1, \dots, Q} (\alpha_{k,i} F_i)$; and determining the preferred measurements set, Ω . Once the preferred measurements set is thus constructed, average round-trip delay, y_k , and delay-jitter, γ_k , are measured between the pair of routers, $\pi_k = (R_i, R_j)$, $k = 1, 2, \dots, Q$, using a network management tool (e.g., ANMT) (step 350). Once these measurements have been obtained, the round-trip delay for router pairs in Φ can be estimated and the round-trip delay vector, Δ_k , constructed (step 360). In a similar fashion, round-trip delay-jitter, σ_k , can be computed for router pairs in Φ (step 370). The process of taking measurements, and estimating and

computing delay and delay-jitter, respectively, can then be repeated for subsequent time intervals, in order to collect more data regarding delay and delay-jitter.

The above process may be summarized as follows:

- 5 BEGIN
 - Specify the requirements set, Φ ;
 - Order the requirements set, Φ ;
 - Identify the N+M delay components and construct the resulting delay-components vector, x ;
 - 10 Determine F matrix from the given requirements set, Φ , and the routing table; Perform "preferred_measurements_set" to:
 - Determine maximal set of independent rows of F, viz. F_1, F_2, \dots, F_Q ;
 - Determine $\alpha_k = [\alpha_{k,1}, \alpha_{k,2}, \dots, \alpha_{k,Q}]$, such that $F_k = \sum_{i=1, \dots, Q} (\alpha_{k,i} F_i)$; and
 - Determine the preferred measurements set, Ω ;
 - 15 Measure round-trip delay and delay-jitter between router pairs in Ω , and organize them as y and c ; respectively
 - Estimate round-trip delay using the equation $\Delta = Ay$;
 - Compute round-trip delay-jitter.
 - END
 - 20

Measurement error is preferably kept to a negligible level.

- Each of the blocks of the flow diagram of Fig. 3, and those depicted in subsequent figures, may be executed by a module (e.g., a software module) or a portion of a module or a computer system user. The methods described herein, the operations thereof and modules for performing such methods may therefore be
 - 25 executed on a computer system configured to execute the operations of the method and/or may be executed from computer-readable media. The method may be embodied in a machine-readable and/or computer-readable medium for configuring a computer system to execute the method. The software modules may be stored within
 - 30 and/or transmitted to a computer system memory to configure the computer system to perform the functions of the module. Alternatively, such actions may be embodied in the structure of circuitry that implements such functionality, such as the micro-code of a complex instruction set computer (CISC), firmware programmed into programmable or erasable/programmable devices, the configuration of a field-programmable gate

array (FPGA), the design of a gate array or full-custom application-specific integrated circuit (ASIC), or the like.

Those skilled in the art will also recognize that the boundaries between modules and operations depicted herein are merely illustrative and alternative
 5 embodiments may merge such modules or operations, or impose an alternative decomposition of functionality thereon. For example, the actions discussed herein may be decomposed into sub-operations to be executed as multiple computer processes. Moreover, alternative embodiments may combine multiple instances of a particular operation or sub-operation. Furthermore, those skilled in the art will
 10 recognize that the operations described in exemplary embodiment are for illustration only. Operations may be combined or the functionality of the operations may be distributed in additional operations in accordance with the invention. As will also be apparent to those of skill in the art, methods for determining delay and jitter described herein may employ other techniques (similar in effect to those described herein) to
 15 make such determinations, and such alternative techniques are intended to be comprehended by the methods and apparatus discussed herein.

A Formalized Description Of A Method For Calculating Delay And Jitter Within A Requirements Set Using A Preferred Measurements Set

A procedure according to a method of the present invention that computes the
 20 round-trip delay and delay-jitter between every pair of routers in the requirements set (using every router pair $\pi_k = (R_i, R_j) \in \Phi$) using measurements between pairs of routers identified by the ordered router pairs of a preferred measurements set, Ω , is now described using a formal syntax. The scheme is particularly useful when the measurement errors are small and no *a priori* delay measurements are available.

25

09633130 "073100

BEGIN

```

/* The following procedure assumes that the topology and routing table, */
/* for a given DS codepoint, is known. Where necessary, this procedure */
5 /* is executed for each DS codepoint under consideration. */
/* If the network topology and/or the routing changes, this procedure */
/* should be executed afresh with the new topology or new routing table. */

/* Specify the collection of pairs of routers */
10  $\Phi_{TEMP} = \{\pi_k = (R_i, R_j), k = 1, 2, \dots, P\};$ 

/* Call order_requirements_set to order the pairs of routers in */
/* the requirements set */
 $\Phi = \text{order\_requirements\_set}(\Phi_{TEMP});$ 
15

/* Identify the N+M delay components, viz. link-level IP transfer delay, */
/*  $d_i, i = 1, 2, \dots, M$ , and ANMT processing delay,  $s_j, j = 1, 2, \dots, N$ , and */
/* Construct the resulting delay-components vector (M-vector, x) */
 $x = [d_1, d_2, \dots, d_M, s_1, s_2, \dots, s_N]^T;$ 
20

For each  $k = 1, 2, \dots, P$ , {
    Identify the pair of routers  $\pi_k = (R_i, R_j), \pi_k \in \Phi;$ 
    Obtain the route  $\text{link\_path}(R_i, R_j) = \langle L_p, L_q, \dots, L_r \rangle;$ 
    Determine the row vector,  $F_k$ , such that  $z_k = \text{Delay}(R_i, R_j) = F_k x$ 
25 };

Construct the  $P \times (N+M)$  matrix, F;

/* Call "preferred_measurements_set" to: */
30 /* Determine the maximal set of independent rows of F, viz.  $F_1, F_2, \dots, F_Q$  */
/* Determine  $\alpha_\kappa = [\alpha_{\kappa,1}, \alpha_{\kappa,2}, \dots, \alpha_{\kappa,Q}]$ , such that  $F_\kappa = \sum_{i=1, \dots, Q} (\alpha_{\kappa,i} F_i)$  */
/* Determine the preferred measurements set,  $\Omega$ . */

( $Q, \Omega, \alpha_\kappa (\kappa = Q+1, \dots, P)$ ) = preferred_measurements_set (F, A,  $\Phi$ );
35

/* Identify the start time,  $t_0$ , and the measurement interval, dt */
 $t_0 = \text{start\_time};$ 
 $dt = \text{measurement\_interval};$ 

40 While (!finished()) {
    if (current-time  $\geq t_0$ )
        For each  $k = 1, 2, \dots, Q$ , {
            Measure average round-trip delay,  $y_k$ , and delay-jitter,
             $\gamma_k$ , between the pair of routers,  $\pi_k = (R_i, R_j)$  using a
45 network management tool (e.g., ANMT)

```

}
 };
 /* Compute round-trip delay vector, Δ_k , for each pair of routers in Φ */
 5 For each router pair in Φ {
 For each $k = 1, 2, \dots, Q$, {
 $\Delta_k = y_k$
 };
 For each $k = Q+1, Q+2, \dots, P$, {
 10 $\Delta_k = \sum_{i=1, \dots, Q} (\alpha_{k,i} y_i)$
 }
 };
 /* Compute round-trip delay-jitter, σ_k , for each pair of routers in Φ */
 15 For each router pair in Φ {
 For each $k = 1, 2, \dots, Q$, {
 $\sigma_k = \gamma_k$
 };
 For each $k = Q+1, Q+2, \dots, P$, {
 20 $\sigma_k = \sqrt{\sum_{i=1, \dots, Q} (\alpha_{k,i}^2 \gamma_i^2)}$
 }
 };
 $t_0 = t_0 + dt$
 }
 25 END

In the above procedure, the call to finished() is made to determine if the number of
 measurements and estimations made are sufficient in the given situation. As noted,
 30 the above scheme is applicable where no other delay measurements or *a priori*
 estimates are available and assumes that the measurement error is negligible.

Operation of order requirements set

Fig. 4 is a flow diagram that describes, in general terms, the operation of
 order_requirements_set. From the given requirements set, Φ , the collection of
 35 routers, \mathcal{R} , for which delay and jitter estimates are required, is identified (step 400).
 Once these routers identified, a preference is assigned to each router (step 410). Next,
 for each router pair in Φ (step 420), the first router of the current router pair is
 compared to the second router of the current router pair (step 430). If the second

router of the current router pair is preferred over the first router of that router pair (step 430), the routers are re-ordered (i.e., swapped) (step 440). Elements $\pi_k = (R_i, R_j) \in \Phi$ are then sorted on the two keys, with regard to $\text{pref}(R_i)$ and $\text{pref}(R_j)$ (step 450), such that the following relationships exists:

5

Assuming $(\pi_{k1} = (R_{i1}, R_{j1}))$ and $\pi_{k2} = (R_{i2}, R_{j2}))$ (38)
 if $(\text{pref}(R_{i1}) \triangleright \text{pref}(R_{i2}))$, then $\text{pref}(R_{i1}) < \text{pref}(R_{i2})$
 if $(\text{pref}(R_{i1}) = \text{pref}(R_{i2}))$, then $\text{pref}(R_{j1}) < \text{pref}(R_{j2})$

10 These operations can be expressed more formally in the following manner.

BEGIN

15 /* From the given requirements set, Φ , identify the collection of */
 /* enterprise routers, \mathfrak{R} , for which delay and jitter estimates are */
 /* required. */

$\mathfrak{R} = \cup R_i$, such that for some R_j , $\pi_k = (R_i, R_j) \in \Phi$ or $\pi_k = (R_j, R_i) \in \Phi$;

20 /* Assign preference to each router */
 For all $R_i \in \mathfrak{R}$ {
 $\text{pref}(R_i) = u, u \in \{1, 2, \dots, 2P\}$
 };

25 /* Re-order each pair of router in Φ */
 For all $\pi_k = (R_i, R_j) \in \Phi$ {
 If $(\text{pref}(R_i) > \text{pref}(R_j))$ {
 $\pi_k = (R_j, R_i)$
 }
 }

Sort elements $\pi_k = (R_i, R_j) \in \Phi$ on the two keys, $\text{pref}(R_i)$ and $\text{pref}(R_j)$, such that the following condition is true:

35

For $(\pi_{k1} = (R_{i1}, R_{j1}))$ and $\pi_{k2} = (R_{i2}, R_{j2})) \&\& (\pi_{k1}, \pi_{k2} \in \Phi)$ {
 if $(\text{pref}(R_{i1}) \triangleright \text{pref}(R_{i2}))$, then $\text{pref}(R_{i1}) < \text{pref}(R_{i2})$;
 if $(\text{pref}(R_{i1}) = \text{pref}(R_{i2}))$, then $\text{pref}(R_{j1}) < \text{pref}(R_{j2})$
 }

40

END

Operation of preferred measurements set

The procedure preferred_measurements_set performs an LU decomposition of matrix F using, for example, Gaussian elimination. The use of Gaussian elimination is merely exemplary, and other methods providing similar results will be apparent to those of skill in the art. Using Gaussian elimination to perform the decomposition, for example, a system of linear algebraic equations, such as

$$\begin{aligned} F_{(1,1)} x_1 + F_{(1,2)} x_2 + F_{(1,3)} x_3 &= b_1 & (39a) \\ F_{(2,1)} x_1 + F_{(2,2)} x_2 + F_{(2,3)} x_3 &= b_2 & (39b) \\ F_{(3,1)} x_1 + F_{(3,2)} x_2 + F_{(3,3)} x_3 &= b_3 & (39c) \end{aligned}$$

can be re-written as

$$F x = b \quad (40)$$

where F is a 3x3 matrix with elements, $F_{(i,j)}$, and x and b are the column vectors

$$x^T = [x_1 \ x_2 \ x_3] \quad (41)$$

$$b^T = [b_1 \ b_2 \ b_3] \quad (42)$$

The set of equations have a unique solution provided that the matrix F is non-singular. In other words, the set of equations have a unique solution provided that the row vectors,

$$F_1^T = [F_{(1,1)} \ F_{(1,2)} \ F_{(1,3)}] \quad (43a)$$

$$F_2^T = [F_{(2,1)} \ F_{(2,2)} \ F_{(2,3)}] \quad (43b)$$

$$F_3^T = [F_{(3,1)} \ F_{(3,2)} \ F_{(3,3)}] \quad (43c)$$

are independent. Once this constraint is met, Gaussian elimination may be used to obtain a 3x3 matrix, A, such that

$$A F = F' \quad (44)$$

where F' is an upper triangular matrix. The fact that F' is an upper triangular matrix implies that the system of equations

$$A F x = F' x = A b \quad (45)$$

can be solved by back substitution. The algorithm to obtain a matrix A (equivalently F') can be described as follows:

```

10      A = I;                                /* where I is the n x n identity matrix */
      for k = 1 to n - 1 {
          for i = k to n {
               $F_i = -(F_{(i,k)} / F_{(k,k)}) F_k + F_i;$       /* These are row vector
                                                              operations. The operation may be
                                                              limited to elements in columns k,
15               $A_i = -(F_{(i,k)} / F_{(k,k)}) A_k + A_i$       /* The operation must be applied
                                                              to elements in columns 1, 2, ..., n
                                                              */
          }
20      }

```

It will be noted that Gaussian elimination may also be used to detect whether the system of equations is solvable, or equivalently, whether the row vectors of F are linearly independent. A dependent row of F can immediately be detected as soon as it is discovered that a pivot element ($F_{(k,k)}$, in the procedure) is zero. In fact, at any point in the process, if $F_{(k,k)} = 0$, then the row F_k is a linear function of F_1, F_2, \dots, F_{k-1} . The dependence between row F_k and F_1, F_2, \dots, F_{k-1} is captured by (and reflected in) the corresponding row in matrix A .

As part of the process of preferred_measurements_set, a sequence of lower-triangular matrices A^1, A^2, \dots, A^k is constructed to obtain a sequence of upper-triangular matrices $F^1 = A^1 F, F^2 = A^2 F, \dots, F^k = A^k F$. If at any stage the j^{th} row of the matrix F^k is zero, i. e. $F_j^k = 0$, for some $j > k$, then F_j is linearly dependent upon F_1, F_2, \dots, F_k because $F_j^k = 0$ implies:

$$A_{(k,1)}^k F_1 + A_{(k,2)}^k F_2 + \dots + A_{(k,k-1)}^k F_{k-1} + F_j = 0 \quad (46)$$

The foregoing forms the basis of the procedure preferred_measurements_set, which is now described in greater detail.

Fig. 5 is a flow diagram that describes, in general terms, the operation of preferred_measurements_set for a given requirements set, $\Phi = \{\pi_{\kappa} = (R_i, R_j), \kappa = 1, 2, \dots, P\}$, and the corresponding $P \times (N+M)$ matrix F . First, the A matrix is set to identity matrix (I) (step 500). For each row up to the last independent row (step 505), the following actions are performed. The next non-zero row of F is identified, if there is such a row that still has not been examined (step 510). If such a row is successfully identified (i.e., $F_j \neq 0$) (step 5015), the rows of F are re-arranged (step 520). Corresponding rows and columns of A are also re-arranged, as are pairs of routers in Φ (steps 525, 530 and 535). At this point in the process, row $F_k \neq 0$. The corresponding non-zero element ($F_{(k,j)} \neq 0$) is then found (step 540). Once the F matrix and X matrix have been prepared for pivoting (by ensuring that pivot $F_{(k,k)} \neq 0$) (step 545), a pivot operation is performed on the F matrix (step 550). A pivot operation is also performed on the A matrix (step 555). At this point the value of Q (the number of independent rows (i.e., rank) of F is known, and the for Q rows of F are independent. Thus, the preferred measurements set can then be populated with the router pairs corresponding to the independent rows of F (step 560). Vectors that allow the non-independent rows of F to be described in terms of the independent rows of F ($\alpha_{\kappa}, \kappa = Q+1, \dots, P$) are then defined (step 565). Finally, values for Q , Ω , and α_{κ} ($\kappa = Q+1, \dots, P$) are returned (step 570). The preceding process can be expressed more formally in the following manner.

BEGIN

```

/* Matrix A is set to the identity matrix (I) */
for i = 1 to P {
5       for j = 1 to P {
           if (i == j) {
               A(i,j) = 1
           }
           else {
10              A(i,j) = 0
           }
       }
    };
    row_zero_at = P+1;
15    k = 1;
    while (k < row_zero_at - 1) {
        j = k;

        /* Once the following is completed, either j has progressed to */
        /* row_zero_at, or Fj is non-zero (i.e., ≠ 0) */
20        while ((Fj = 0) && (j < row_zero_at)) {
            j = j + 1
        };

25        if ((Fj > 0) && (j < row_zero_at)) {

            /* Rearrange rows of F, and (correspondingly) rows and */
            /* columns of A, and pairs of routers in Φ. */
            /* (1) interchange_matrix_rows (MATRIX, FROM, TO) */
            /* exchanges the FROM and TO rows of MATRIX */
30            /* (2) interchange_matrix_columns (MATRIX, FROM, TO) */
            /* exchanges corresponding columns of MATRIX */
            /* (3) interchange_router_pairs (SET, I, J) */
            /* exchanges router pairs I & J */

35            to_row = k;
            for i = j to (row_zero_at - 1) {
                interchange_matrix_rows (F, i, to_row);
                interchange_matrix_rows (A, i, to_row);
                interchange_matrix_columns (A, i, to_row);
                interchange_router_pairs (Φ, i, to_row);
40                to_row = to_row + 1
            };
            /* At this stage, row Fk ≠ 0. */

```

09638180-073100
007E20-02B5950


```

row_zero_at = to_row;
j = k;
while  $F_{(k,j)} = 0$  {
    j = j+1                                /* At this stage,  $F_{(k,j)} \neq 0$ . */
};

/* The following ensures that pivot  $F_{(k,k)} \neq 0$ . */
/* Once completed, pivoting may be carried out. */
interchange_matrix_columns(F, k, j);
interchange_vector_elements(x, k, j);

/* Undertake pivot operation */
if k < row_zero_at then
    for all i = k+1 to row_zero_at - 1 {
        /* These are row vector operations. */
        /* The operation may be limited to elements in */
        /* columns k, k+1, ..., M */
         $F_i = -(F_{(i,k)} / F_{(k,k)}) F_k + F_i$ ;

        /* The operation must be applied to elements in */
        /* columns 1, 2, ..., P */
         $A_i = -(F_{(i,k)} / F_{(k,k)}) A_k + A_i$ ;
        /* This completes the pivot operation. */

        k = k+1
    }
else {                                /* i.e., for the remaining  $F_j = 0$ . */
    row_zero_at = k
}

/*      End of outer while loop */

Q = row_zero_at - 1;
 $\Omega = \{\}$ ;
for  $\kappa = 1$  to Q {
    add_to_set( $\Omega$ ,  $\pi_\kappa$ )
};

for  $\kappa = Q+1$  to P {                    /* Req'd to ensure  $F_\kappa = \sum_{i=1, \dots, Q} (\alpha_{\kappa,i} F_i)$  */
     $\alpha_\kappa = [-A_{(\kappa,1)}, -A_{(\kappa,2)}, \dots, -A_{(\kappa,Q)}]$ 
};

return(Q,  $\Omega$ ,  $\alpha_\kappa$ ,  $\kappa = Q+1, \dots, P$ )

```

END

A Measurement Example Using a Preferred Measurements Set

Fig. 6 is a block diagram depicting an exemplary enterprise network 600 in which a measurement and estimation technique according to the present invention can be employed to estimate delay and delay-jitter. Enterprise network 600 includes a backbone 610 and routers 620(1)-(7). Routers 620(1)-(7) are coupled to backbone 610 by links 630(1)-(7), respectively. It will be noted that routers 620(1)-(7) are also designated R_i , $i = 1, 2, \dots, 7$, and that links 630(1)-(7) are also designated $L_{(i,j)}$, $i = 1, 2, 3, 4$, and $j = 1, 2, 3$, in order to allow network 600 to be discussed in terms of the parameters discussed herein. Additionally, backbone 610 includes backbone routers 640(1)-(4), with backbone router 640(1) coupled to backbone routers 640(2), 640(3) and 640(4) by backbone links 650(1), 650(2) and 650(3), respectively. It will be noted that backbone routers 640(1)-(4) are also designated ρ_i , $i = 1, 2, 3, 4$, respectively, and that backbone links 650(1)-(3) are also designated $\lambda_{(i,j)}$, $j = 1, 2, 3$, respectively, in order to allow network 600 to be discussed in terms of the parameters discussed herein. It is desirable to determine, or at least estimate, the delay and delay-jitter experienced by network traffic flowing through enterprise network 600. This information may be derived using the techniques just described.

To begin, let the requirements set be

$$\Phi = \{(R_1, R_2), (R_1, R_3), (R_1, R_4), (R_1, R_5), (R_1, R_6), (R_1, R_7), (R_2, R_3), (R_2, R_4), (R_2, R_5), (R_2, R_6), (R_2, R_7), (R_3, R_4), (R_3, R_5), (R_3, R_6), (R_3, R_7)\} \quad (47)$$

The various actions performed when estimating delay and delay-jitter using a preferred measurements set are now described.

First, the operator assigns priorities to one or more of the routers in the collection of enterprise routers, $\mathfrak{R} = \{R_1, R_2, R_3, R_4, R_5, R_6, R_7\}$. For example, the assigned priorities can be, in decreasing priority, $R_1, R_4, R_5, R_6, R_2, R_3, R_7$. The pairs of routers are re-ordered as follows:

5

$$\Phi = \{(R_1, R_2), (R_1, R_3), (R_1, R_4), (R_1, R_5), (R_1, R_6), (R_1, R_7), (R_2, R_3), (R_4, R_2), (R_5, R_2), (R_6, R_2), (R_2, R_7), (R_4, R_3), (R_5, R_3), (R_6, R_3), (R_3, R_7)\} \quad (48)$$

10 The set, Φ , when sorted on the two keys, $\text{pref}(R_i)$ and $\text{pref}(R_j)$ results in:

$$\Phi = \{(R_1, R_2), (R_1, R_3), (R_1, R_4), (R_1, R_5), (R_1, R_6), (R_1, R_7), (R_4, R_2), (R_4, R_3), (R_5, R_2), (R_5, R_3), (R_6, R_2), (R_6, R_3), (R_2, R_3), (R_2, R_7), (R_3, R_7)\} \quad (49)$$

15

Next, the N+M-vector, x , of delay components is determined

$$x = [d_{11}, d_{12}, d_{13}, d_{21}, d_{31}, d_{41}, d_{42}, c_{12}, c_{13}, c_{14}, s_{11}, s_{12}, s_{13}, s_{21}, s_{31}, s_{41}, s_{42}]^T \quad (50)$$

20 The resulting F matrix is a 15x17 matrix of the form:

	d11	d12	d13	d21	d31	d41	d42	C12	C13	C14	s11	s12	s13	s21	s31	s41	s42
y1	1	1						0	0	0	1	1	0	0	0	0	0
y2	1		1					0	0	0	1	0	1	0	0	0	0
y3	1			1				1	0	0	1	0	0	1	0	0	0
y4	1				1			0	1	0	1	0	0	0	1	0	0
y5	1					1		0	0	1	1	0	0	0	0	1	0
y6	1						1	0	0	1	1	0	0	0	0	0	1
y7		1		1				1	0	0	0	1	0	1	0	0	0
y8			1	1				1	0	0	0	0	1	1	0	0	0
y9		1			1			0	1	0	0	1	0	0	1	0	0
y10			1		1			0	1	0	0	0	1	0	1	0	0
y11		1				1		0	0	1	0	1	0	0	0	1	0
y12			1			1		0	0	1	0	0	1	0	0	1	0
y13		1	1					0	0	0	0	1	1	0	0	0	0
y14		1					1	0	0	1	0	1	0	0	0	0	1
y15			1				1	0	0	1	0	0	1	0	0	0	1

Initially, the pivot element is F(1,1). As a result matrix F is:

	d11	d12	d13	d21	d31	d41	d42	C12	C13	C14	s11	s12	s13	s21	s31	s41	s42
y1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
y2	0	-1	1	0	0	0	0	0	0	0	0	-1	1	0	0	0	0
y3	0	-1	0	1	0	0	0	1	0	0	0	-1	0	1	0	0	0
y4	0	-1	0	0	1	0	0	0	1	0	0	-1	0	0	1	0	0
y5	0	-1	0	0	0	1	0	0	0	1	0	-1	0	0	0	1	0
y6	0	-1	0	0	0	0	1	0	0	1	0	-1	0	0	0	0	1
y7	0	1	0	1	0	0	0	1	0	0	0	1	0	1	0	0	0
y8	0	0	1	1	0	0	0	1	0	0	0	0	1	1	0	0	0
y9	0	1	0	0	1	0	0	0	1	0	0	1	0	0	1	0	0
y10	0	0	1	0	1	0	0	0	1	0	0	0	1	0	1	0	0
y11	0	1	0	0	0	1	0	0	0	1	0	1	0	0	0	1	0
y12	0	0	1	0	0	1	0	0	0	1	0	0	1	0	0	1	0
y13	0	1	1	0	0	0	0	0	0	0	0	1	1	0	0	0	0
y14	0	1	0	0	0	0	1	0	0	1	0	1	0	0	0	0	1
y15	0	0	1	0	0	0	1	0	0	1	0	0	1	0	0	0	1

The next pivot element is F(2,2). As a result matrix F is:

5

	d11	d12	d13	d21	d31	d41	d42	C12	C13	C14	s11	s12	s13	s21	s31	s41	s42
y1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
y2	0	-1	1	0	0	0	0	0	0	0	0	-1	1	0	0	0	0
y3	0	0	-1	1	0	0	0	1	0	0	0	0	-1	1	0	0	0
y4	0	0	-1	0	1	0	0	0	1	0	0	0	-1	0	1	0	0
y5	0	0	-1	0	0	1	0	0	0	1	0	0	-1	0	0	1	0
y6	0	0	-1	0	0	0	1	0	0	1	0	0	-1	0	0	0	1
y7	0	0	1	1	0	0	0	1	0	0	0	0	1	1	0	0	0
y8	0	0	1	1	0	0	0	1	0	0	0	0	1	1	0	0	0
y9	0	0	1	0	1	0	0	0	1	0	0	0	1	0	1	0	0
y10	0	0	1	0	1	0	0	0	1	0	0	0	1	0	1	0	0
y11	0	0	1	0	0	1	0	0	0	1	0	0	1	0	0	1	0
y12	0	0	1	0	0	1	0	0	0	1	0	0	1	0	0	1	0
y13	0	0	2	0	0	0	0	0	0	0	0	2	0	0	0	0	0
y14	0	0	1	0	0	0	1	0	0	1	0	0	1	0	0	0	1
y15	0	0	1	0	0	0	1	0	0	1	0	0	1	0	0	0	1

The next pivot element is F(3,3). As a result matrix F is:

	d11	d12	d13	d21	d31	d41	d42	C12	C13	C14	s11	s12	s13	s21	s31	s41	s42
y1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
y2	0	-1	1	0	0	0	0	0	0	0	0	-1	1	0	0	0	0
y3	0	0	-1	1	0	0	0	1	0	0	0	0	-1	1	0	0	0
y4	0	0	0	-1	1	0	0	-1	1	0	0	0	0	-1	1	0	0
y5	0	0	0	-1	0	1	0	-1	0	1	0	0	0	-1	0	1	0
y6	0	0	0	-1	0	0	1	-1	0	1	0	0	0	-1	0	0	1
y7	0	0	0	2	0	0	0	2	0	0	0	0	0	2	0	0	0
y8	0	0	0	2	0	0	0	2	0	0	0	0	0	2	0	0	0
y9	0	0	0	1	1	0	0	1	1	0	0	0	0	1	1	0	0
y10	0	0	0	1	1	0	0	1	1	0	0	0	0	1	1	0	0
y11	0	0	0	1	0	1	0	1	0	1	0	0	0	1	0	1	0
y12	0	0	0	1	0	1	0	1	0	1	0	0	0	1	0	1	0
y13	0	0	0	2	0	0	0	2	0	0	0	0	0	2	0	0	0
y14	0	0	0	1	0	0	1	1	0	1	0	0	0	1	0	0	1
y15	0	0	0	1	0	0	1	1	0	1	0	0	0	1	0	0	1

The next pivot element is F(4,4). As a result matrix F is:

	d11	d12	d13	d21	d31	d41	d42	C12	C13	C14	s11	s12	s13	s21	s31	s41	s42
y1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
y2	0	-1	1	0	0	0	0	0	0	0	0	-1	1	0	0	0	0
y3	0	0	-1	1	0	0	0	1	0	0	0	0	-1	1	0	0	0
y4	0	0	0	-1	1	0	0	-1	1	0	0	0	0	-1	1	0	0
y5	0	0	0	0	-1	1	0	0	-1	1	0	0	0	0	-1	1	0
y6	0	0	0	0	-1	0	1	0	-1	1	0	0	0	0	-1	0	1
y7	0	0	0	0	2	0	0	0	2	0	0	0	0	0	2	0	0
y8	0	0	0	0	2	0	0	0	2	0	0	0	0	0	2	0	0
y9	0	0	0	0	2	0	0	0	2	0	0	0	0	0	2	0	0
y10	0	0	0	0	2	0	0	0	2	0	0	0	0	0	2	0	0
y11	0	0	0	0	1	1	0	0	1	1	0	0	0	0	1	1	0
y12	0	0	0	0	1	1	0	0	1	1	0	0	0	0	1	1	0
y13	0	0	0	0	2	0	0	0	2	0	0	0	0	0	2	0	0
y14	0	0	0	0	1	0	1	0	1	1	0	0	0	0	1	0	1
y15	0	0	0	0	1	0	1	0	1	1	0	0	0	0	1	0	1

5

The next pivot element is F(5,5). As a result matrix F is:

	d11	d12	d13	d21	d31	d41	d42	C12	C13	C14	s11	s12	s13	s21	s31	s41	s42
y1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
y2	0	-1	1	0	0	0	0	0	0	0	0	0	-1	1	0	0	0
y3	0	0	-1	1	0	0	0	1	0	0	0	0	0	-1	1	0	0
y4	0	0	0	-1	1	0	0	-1	1	0	0	0	0	-1	1	0	0
y5	0	0	0	0	-1	1	0	0	-1	1	0	0	0	0	-1	1	0
y6	0	0	0	0	0	-1	1	0	0	0	0	0	0	0	0	-1	1
y7	0	0	0	0	0	2	0	0	0	2	0	0	0	0	0	2	0
y8	0	0	0	0	0	2	0	0	0	2	0	0	0	0	0	2	0
y9	0	0	0	0	0	2	0	0	0	2	0	0	0	0	0	2	0
y10	0	0	0	0	0	2	0	0	0	2	0	0	0	0	0	2	0
y11	0	0	0	0	0	2	0	0	0	2	0	0	0	0	0	2	0
y12	0	0	0	0	0	2	0	0	0	2	0	0	0	0	0	2	0
y13	0	0	0	0	0	2	0	0	0	2	0	0	0	0	0	2	0
y14	0	0	0	0	0	1	1	0	0	2	0	0	0	0	0	1	1
y15	0	0	0	0	0	1	1	0	0	2	0	0	0	0	0	1	1

The next pivot element is F(6,6). As a result matrix F is:

	d11	d12	d13	d21	d31	d41	d42	C12	C13	C14	s11	s12	s13	s21	s31	s41	s42
y1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
y2	0	-1	1	0	0	0	0	0	0	0	0	0	-1	1	0	0	0
y3	0	0	-1	1	0	0	0	1	0	0	0	0	0	-1	1	0	0
y4	0	0	0	-1	1	0	0	-1	1	0	0	0	0	-1	1	0	0
y5	0	0	0	0	-1	1	0	0	-1	1	0	0	0	0	-1	1	0
y6	0	0	0	0	0	-1	1	0	0	0	0	0	0	0	0	-1	1
y7	0	0	0	0	0	0	2	0	0	2	0	0	0	0	0	0	2
y8	0	0	0	0	0	0	2	0	0	2	0	0	0	0	0	0	2
y9	0	0	0	0	0	0	2	0	0	2	0	0	0	0	0	0	2
y10	0	0	0	0	0	0	2	0	0	2	0	0	0	0	0	0	2
y11	0	0	0	0	0	0	2	0	0	2	0	0	0	0	0	0	2
y12	0	0	0	0	0	0	2	0	0	2	0	0	0	0	0	0	2
y13	0	0	0	0	0	0	2	0	0	2	0	0	0	0	0	0	2
y14	0	0	0	0	0	0	2	0	0	2	0	0	0	0	0	0	2
y15	0	0	0	0	0	0	2	0	0	2	0	0	0	0	0	0	2

The next pivot element is F(7,7). As a result matrix F is:

	d11	d12	d13	d21	d31	d41	d42	C12	C13	C14	s11	s12	s13	s21	s31	s41	s42
y1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
y2	0	-1	1	0	0	0	0	0	0	0	0	-1	1	0	0	0	0
y3	0	0	-1	1	0	0	0	1	0	0	0	0	-1	1	0	0	0
y4	0	0	0	-1	1	0	0	-1	1	0	0	0	0	-1	1	0	0
y5	0	0	0	0	-1	1	0	0	-1	1	0	0	0	0	-1	1	0
y6	0	0	0	0	0	-1	1	0	0	0	0	0	0	0	0	-1	1
y7	0	0	0	0	0	0	2	0	0	2	0	0	0	0	0	0	2
y8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
y9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
y10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
y11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
y12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
y13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
y14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
y15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Since the remaining row vectors of F are 0, the algorithm terminates.

5 The sequence of resulting 15x15 A matrices are given below. Initially, the A matrix is:

	y1	y2	y3	y4	y5	y6	y7	y8	y9	y10	y11	y12	y13	y14	y15
y1	1														
y2		1													
y3			1												
y4				1											
y5					1										
y6						1									
y7							1								
y8								1							
y9									1						
y10										1					
y11											1				
y12												1			
y13													1		
y14														1	
y15															1

Using the pivot F(1,1), matrix A is:

	y1	y2	y3	y4	y5	y6	y7	y8	y9	y10	y11	y12	y13	y14	y15
y1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
y2	-1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
y3	-1	0	1	0	0	0	0	0	0	0	0	0	0	0	0
y4	-1	0	0	1	0	0	0	0	0	0	0	0	0	0	0
y5	-1	0	0	0	1	0	0	0	0	0	0	0	0	0	0
y6	-1	0	0	0	0	1	0	0	0	0	0	0	0	0	0
y7	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
y8	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
y9	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
y10	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
y11	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
y12	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
y13	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
y14	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
y15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Using the pivot F(2,2), matrix A is:

5

	y1	y2	y3	y4	y5	y6	y7	y8	y9	y10	y11	y12	y13	y14	y15
y1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
y2	-1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
y3	0	-1	1	0	0	0	0	0	0	0	0	0	0	0	0
y4	0	-1	0	1	0	0	0	0	0	0	0	0	0	0	0
y5	0	-1	0	0	1	0	0	0	0	0	0	0	0	0	0
y6	0	-1	0	0	0	1	0	0	0	0	0	0	0	0	0
y7	-1	1	0	0	0	0	1	0	0	0	0	0	0	0	0
y8	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
y9	-1	1	0	0	0	0	0	0	1	0	0	0	0	0	0
y10	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
y11	-1	1	0	0	0	0	0	0	0	0	1	0	0	0	0
y12	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
y13	-1	1	0	0	0	0	0	0	0	0	0	0	1	0	0
y14	-1	1	0	0	0	0	0	0	0	0	0	0	0	1	0
y15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Using the pivot F(3,3), matrix A is:

	y1	y2	y3	y4	y5	y6	y7	y8	y9	y10	y11	y12	y13	y14	y15
y1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
y2	-1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
y3	0	-1	1	0	0	0	0	0	0	0	0	0	0	0	0
y4	0	0	-1	1	0	0	0	0	0	0	0	0	0	0	0
y5	0	0	-1	0	1	0	0	0	0	0	0	0	0	0	0
y6	0	0	-1	0	0	1	0	0	0	0	0	0	0	0	0
y7	-1	0	1	0	0	0	1	0	0	0	0	0	0	0	0
y8	0	-1	1	0	0	0	0	1	0	0	0	0	0	0	0
y9	-1	0	1	0	0	0	0	0	1	0	0	0	0	0	0
y10	0	-1	1	0	0	0	0	0	0	1	0	0	0	0	0
y11	-1	0	1	0	0	0	0	0	0	0	1	0	0	0	0
y12	0	-1	1	0	0	0	0	0	0	0	0	1	0	0	0
y13	-1	-1	2	0	0	0	0	0	0	0	0	0	1	0	0
y14	-1	0	1	0	0	0	0	0	0	0	0	0	0	1	0
y15	0	-1	1	0	0	0	0	0	0	0	0	0	0	0	1

Using the pivot F(4,4), matrix A is:

5

	y1	y2	y3	y4	y5	y6	y7	y8	y9	y10	y11	y12	y13	y14	y15
y1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
y2	-1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
y3	0	-1	1	0	0	0	0	0	0	0	0	0	0	0	0
y4	0	0	-1	1	0	0	0	0	0	0	0	0	0	0	0
y5	0	0	0	-1	1	0	0	0	0	0	0	0	0	0	0
y6	0	0	0	-1	0	1	0	0	0	0	0	0	0	0	0
y7	-1	0	-1	2	0	0	1	0	0	0	0	0	0	0	0
y8	0	-1	-1	2	0	0	0	1	0	0	0	0	0	0	0
y9	-1	0	0	1	0	0	0	0	1	0	0	0	0	0	0
y10	0	-1	0	1	0	0	0	0	0	1	0	0	0	0	0
y11	-1	0	0	1	0	0	0	0	0	0	1	0	0	0	0
y12	0	-1	0	1	0	0	0	0	0	0	0	1	0	0	0
y13	-1	-1	0	2	0	0	0	0	0	0	0	0	1	0	0
y14	-1	0	0	1	0	0	0	0	0	0	0	0	0	1	0
y15	0	-1	0	1	0	0	0	0	0	0	0	0	0	0	1

Using the pivot F(5,5), matrix A is:

	y1	y2	y3	y4	y5	y6	y7	y8	y9	y10	y11	y12	y13	y14	y15
y1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
y2	-1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
y3	0	-1	1	0	0	0	0	0	0	0	0	0	0	0	0
y4	0	0	-1	1	0	0	0	0	0	0	0	0	0	0	0
y5	0	0	0	-1	1	0	0	0	0	0	0	0	0	0	0
y6	0	0	0	0	-1	1	0	0	0	0	0	0	0	0	0
y7	-1	0	-1	0	2	0	1	0	0	0	0	0	0	0	0
y8	0	-1	-1	0	2	0	0	1	0	0	0	0	0	0	0
y9	-1	0	0	-1	2	0	0	0	1	0	0	0	0	0	0
y10	0	-1	0	-1	2	0	0	0	0	1	0	0	0	0	0
y11	-1	0	0	0	1	0	0	0	0	0	1	0	0	0	0
y12	0	-1	0	0	1	0	0	0	0	0	0	1	0	0	0
y13	-1	-1	0	0	2	0	0	0	0	0	0	0	1	0	0
y14	-1	0	0	0	1	0	0	0	0	0	0	0	0	1	0
y15	0	-1	0	0	1	0	0	0	0	0	0	0	0	0	1

Using the pivot F(6,6), matrix A is:

5

	y1	y2	y3	y4	y5	y6	y7	y8	y9	y10	y11	y12	y13	y14	y15
y1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
y2	-1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
y3	0	-1	1	0	0	0	0	0	0	0	0	0	0	0	0
y4	0	0	-1	1	0	0	0	0	0	0	0	0	0	0	0
y5	0	0	0	-1	1	0	0	0	0	0	0	0	0	0	0
y6	0	0	0	0	-1	1	0	0	0	0	0	0	0	0	0
y7	-1	0	-1	0	0	2	1	0	0	0	0	0	0	0	0
y8	0	-1	-1	0	0	2	0	1	0	0	0	0	0	0	0
y9	-1	0	0	-1	0	2	0	0	1	0	0	0	0	0	0
y10	0	-1	0	-1	0	2	0	0	0	1	0	0	0	0	0
y11	-1	0	0	0	-1	2	0	0	0	0	1	0	0	0	0
y12	0	-1	0	0	-1	2	0	0	0	0	0	1	0	0	0
y13	-1	-1	0	0	0	2	0	0	0	0	0	0	1	0	0
y14	-1	0	0	0	0	1	0	0	0	0	0	0	0	1	0
y15	0	-1	0	0	0	1	0	0	0	0	0	0	0	0	1

Using the pivot F(7,7), matrix A is:

	y1	y2	y3	y4	y5	y6	y7	y8	y9	y10	y11	y12	y13	y14	y15
y1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
y2	-1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
y3	0	-1	1	0	0	0	0	0	0	0	0	0	0	0	0
y4	0	0	-1	1	0	0	0	0	0	0	0	0	0	0	0
y5	0	0	0	-1	1	0	0	0	0	0	0	0	0	0	0
y6	0	0	0	0	-1	1	0	0	0	0	0	0	0	0	0
y7	-1	0	-1	0	0	2	1	0	0	0	0	0	0	0	0
y8	1	-1	0	0	0	0	-1	1	0	0	0	0	0	0	0
y9	0	0	1	-1	0	0	-1	0	1	0	0	0	0	0	0
y10	1	-1	1	-1	0	0	-1	0	0	1	0	0	0	0	0
y11	0	0	1	0	-1	0	-1	0	0	0	1	0	0	0	0
y12	1	-1	1	0	-1	0	-1	0	0	0	0	1	0	0	0
y13	0	-1	1	0	0	0	-1	0	0	0	0	0	1	0	0
y14	0	0	1	0	0	-1	-1	0	0	0	0	0	0	1	0
y15	1	-1	1	0	0	-1	-1	0	0	0	0	0	0	0	1

Thus, using the procedure preferred_measurements_set, the resulting 15x17 matrix F'

5 is:

	d11	d12	d13	d21	d31	d41	d42	C12	C13	C14	s11	s12	s13	s21	s31	s41	s42
y1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
y2	0	-1	1	0	0	0	0	0	0	0	0	-1	1	0	0	0	0
y3	0	0	-1	1	0	0	0	1	0	0	0	0	-1	1	0	0	0
y4	0	0	0	-1	1	0	0	-1	1	0	0	0	0	-1	1	0	0
y5	0	0	0	0	-1	1	0	0	-1	1	0	0	0	0	-1	1	0
y6	0	0	0	0	0	-1	1	0	0	0	0	0	0	0	0	-1	1
y7	0	0	0	0	0	0	2	0	0	2	0	0	0	0	0	0	2
y8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
y9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
y10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
y11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
y12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
y13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
y14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
y15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	y1	y2	y3	y4	y5	y6	y7	y8	y9	y10	y11	y12	y13	y14	y15
y1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
y2	-1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
y3	0	-1	1	0	0	0	0	0	0	0	0	0	0	0	0
y4	0	0	-1	1	0	0	0	0	0	0	0	0	0	0	0
y5	0	0	0	-1	1	0	0	0	0	0	0	0	0	0	0
y6	0	0	0	0	-1	1	0	0	0	0	0	0	0	0	0
y7	-1	0	-1	0	0	2	1	0	0	0	0	0	0	0	0
y8	1	-1	0	0	0	0	-1	1	0	0	0	0	0	0	0
y9	0	0	1	-1	0	0	-1	0	1	0	0	0	0	0	0
y10	1	-1	1	-1	0	0	-1	0	0	1	0	0	0	0	0
y11	0	0	1	0	-1	0	-1	0	0	0	1	0	0	0	0
y12	1	-1	1	0	-1	0	-1	0	0	0	0	1	0	0	0
y13	0	-1	1	0	0	0	-1	0	0	0	0	0	1	0	0
y14	0	0	1	0	0	-1	-1	0	0	0	0	0	0	1	0
y15	1	-1	1	0	0	-1	-1	0	0	0	0	0	0	0	1

5 preferred measurements set is:

$$\Omega = \{(R_1, R_2), (R_1, R_3), (R_1, R_4), (R_1, R_5), (R_1, R_6), (R_1, R_7), (R_4, R_2)\} \quad (51)$$

10

	y1	y2	y3	y4	y5	y6	y7
α_8	-1	1	0	0	0	0	1
α_9	0	0	-1	1	0	0	1
α_{10}	-1	1	-1	1	0	0	1
α_{11}	0	0	-1	0	1	0	1
α_{12}	-1	1	-1	0	1	0	1
α_{13}	0	1	-1	0	0	0	1
α_{14}	0	0	-1	0	0	1	1
α_{15}	-1	1	-1	0	0	1	1

- 51 -

Many other examples will be apparent to one of skill in the art. For example, if the assigned priorities are instead, in decreasing priority, $R_4, R_5, R_6, R_7, R_1, R_2, R_3$, then the set, Φ , when sorted on the two keys, $\text{pref}(R_i)$ and $\text{pref}(R_j)$ results in:

$$\Phi = \{(R_4, R_1), (R_4, R_2), (R_4, R_3), (R_5, R_1), (R_5, R_2), (R_5, R_3), (R_6, R_1), (R_6, R_2), (R_6, R_3), (R_7, R_1), (R_7, R_2), (R_7, R_3), (R_1, R_2), (R_1, R_3), (R_2, R_3)\} \quad (52)$$

The corresponding matrix F is:

10

	d11	d12	d13	d21	d31	d41	d42	C12	C13	C14	s11	s12	s13	s21	s31	s41	s42
	1	2	3	4	5	6	7										
y1	1			1				1	0	0	1				1		
y2		1		1				1	0	0		1			1		
y3			1	1				1	0	0			1	1			
y4	1				1			0	1	0	1					1	
y5		1			1			0	1	0		1				1	
y6			1		1			0	1	0			1		1		
y7	1					1		0	1	1	1					1	
y8		1				1		0	1	1		1				1	
y9			1			1		0	1	1			1			1	
y10	1						1	0	0	1	1						1
y11		1					1	0	0	1		1					1
y12			1				1	0	0	1			1				1
y13	1	1						0	0	0	1	1					
y14	1		1					0	0	0	1		1				
y15		1	1					0	0	0		1	1				

Which when decomposed results in:

	d11	d12	d13	d21	d31	d41	d42	C12	C13	C14	s11	s12	s13	s21	s31	s41	s42
y1	1	0	0	1	0	0	0	1	0	0	1	0	0	1	0	0	0
y2	0	1	0	1	0	0	0	1	0	0	0	1	0	1	0	0	0
y3	0	0	1	1	0	0	0	1	0	0	0	0	1	1	0	0	0
y4	0	0	0	-1	1	0	0	-1	1	0	0	0	0	-1	1	0	0
y7	0	0	0	0	-1	1	0	0	0	1	0	0	0	0	-1	1	0
y10	0	0	0	0	0	-1	1	0	-1	0	0	0	0	0	0	-1	1
y13	0	0	0	0	0	0	-2	0	0	-2	0	0	0	0	0	0	-2
Y14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Y15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
y11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
y12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Y8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Y9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Y5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Y6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

15 This establishes the fact that row vectors, $F_1, F_2, F_3, F_4, F_7, F_{10}$, and F_{13} are independent. Consequently, the preferred measurements set is:

$$\Omega = \{(R_4, R_1), (R_4, R_2), (R_4, R_3), (R_5, R_1), (R_6, R_1), (R_7, R_1), (R_1, R_2)\} \quad (53)$$

As can be seen, the resulting preferred measurements set can be controlled, at least to some degree, by prioritizing the appropriate routers. Thus, by careful selection of
 5 routers, such a method and apparatus addresses the operator's (and network's) needs with regard to controlling the routers and links involved in the actual measurement of delay and delay-jitter.

While particular embodiments of the present invention have been shown and described, it will be obvious to those skilled in the art that, based upon the teachings
 10 herein, changes and modifications may be made without departing from this invention and its broader aspects and, therefore, the appended claims are to encompass within their scope all such changes and modifications as are within the true spirit and scope of this invention. Furthermore, it is to be understood that the invention is solely defined by the appended claims.

WHAT IS CLAIMED IS:

1 1. A method of determining a network performance metric in a network,
2 wherein said network comprises a plurality of network elements and each one of said
3 network elements is coupled to at least one other of said network elements by at least
4 one of a plurality of links, comprising:
5 forming a first set of network element pairs, said first set of network element
6 pairs comprising a plurality of pairs of said network elements;
7 ordering a first plurality of network element pairs comprising ones of network
8 element pairs in said first set of network element pairs;
9 forming a second set of network element pairs, wherein said second set of
10 network element pairs comprises ones of said network element pairs in
11 said first set of network element pairs;
12 measuring a measured network performance metric between a first network
13 element pair, wherein said first network element pair comprises a first
14 network element and a second network element of one of said network
15 element pairs in said second set of network element pairs; and
16 computing a computed network performance metric between a second network
17 element pair using said measured network performance metric, wherein
18 said second network element pair comprises a first network element
19 and a second network element of said network element pair in said first
20 set of network element pairs.

1 2. The method of claim 1, wherein said computed network performance
2 metric is computed using a relationship between said first and said second network
3 element pairs.

1 3. The method of claim 1, wherein said ordering comprises:
2 identifying a plurality of network elements, wherein each one of said plurality
3 of network elements is one of a network element pair in said first set of
4 network element pairs;

5 assigning one of a plurality of preferences to each one of said plurality of
 6 network elements; and
 7 sorting said network element pairs in said first set of network element pairs
 8 based on said plurality of preferences.

1 4. The method of claim 3, wherein said sorting comprises:
 2 for each one of said network element pairs in said first set of network element
 3 pairs, swapping a first network element and a second network element
 4 in said each one of said network element pairs in said first set of
 5 network element pairs, if a preference of said first network element in
 6 said each one of said network element pairs in said first set of network
 7 element pairs is less than a preference of said second network element
 8 in said each one of said network element pairs in said first set of
 9 network element pairs;
 10 sorting said network element pairs in said first set of network element pairs
 11 based on a preference of a present first network element of said each
 12 one of said network element pairs in said first set of network element
 13 pairs; and
 14 sorting said network element pairs in said first set of network element pairs
 15 based on a preference of a present second network element of said each
 16 one of said network element pairs in said first set of network element
 17 pairs.

1 5. The method of claim 1, further comprising:
 2 forming a first matrix, wherein each row in said first matrix corresponds to a
 3 network element pair of said first set of network element pairs; and
 4 determining a set of independent rows of said first matrix.

1 6. The method of claim 5, wherein said set of independent rows of said
 2 first matrix is a maximal set of independent rows of said first matrix.

1 7. The method of claim 5, wherein said forming said second set of
 2 network element pairs comprises:
 3 including independent network element pairs in said second set of network
 4 element pairs, wherein
 5 said independent network element pairs are ones of said network
 6 element pairs in said first set of network element pairs
 7 corresponding to rows of said first matrix in said set of
 8 independent rows of said first matrix.

1 8. The method of claim 1, further comprising:
 2 forming a delay components vector;
 3 forming a first matrix, wherein
 4 said first matrix describes a relationship between said delay
 5 components vector and a delay between each of said network
 6 element pairs of said first set of network element pairs,
 7 each row in said first matrix corresponds to a network element pair of
 8 said first set of network element pairs; and
 9 determining a set of independent rows of said first matrix, wherein said
 10 forming said second set of network element pairs comprises
 11 including independent network element pairs in said second set of
 12 network element pairs, and
 13 said independent network element pairs are ones of said network
 14 element pairs in said first set of network element pairs
 15 corresponding to rows of said first matrix in said set of
 16 independent rows of said first matrix.

1 9. The method of claim 8, wherein said delay components vector
2 comprises:
3 a representation of a delay within each network element of each network
4 element pair of said first set of network element pairs for said each
5 network element pair of said first set of network element pairs, and
6 a representation of a delay between network elements of said each network
7 element pair of said first set of network element pairs for said each
8 network element pair of said first set of network element pairs.

1 10. The method of claim 8, further comprising:
2 forming a second matrix, wherein
3 said second matrix describes a relationship between a plurality of
4 independent delays and a non-independent delay,
5 said plurality of independent delays comprise a delay between network
6 elements in each network element pair of said second set of
7 network element pairs, and
8 said non-independent delay comprises a delay between network
9 elements in a network element pair of said first set of network
10 element pairs that is not in said second set of network element
11 pairs.

1 11. The method of claim 10, wherein said forming said second matrix
2 comprises performing a Gaussian elimination using said first and said second
3 matrices.

1 12. The method of claim 10, wherein said computing said computed
2 network performance metric comprises:
3 multiplying said measured performance metric by an element of said second
4 matrix.

096338180 073400

13. A computer system comprising:

a processor;

a network interface, coupled to said processor and to a network, wherein said network comprises a plurality of network elements and each one of said network elements is coupled to at least one other of said network elements by at least one of a plurality of links;

computer readable medium coupled to said processor; and

computer code, encoded in said computer readable medium, configured to cause said processor to:

form a first set of network element pairs, said first set of network element pairs comprising a plurality of pairs of said network elements;

order a first plurality of network element pairs comprising ones of network element pairs in said first set of network element pairs;

form a second set of network element pairs, wherein said second set of network element pairs comprises ones of said network element pairs in said first set of network element pairs;

measure a measured network performance metric between a first network element pair, wherein said first network element pair comprises a first network element and a second network element of one of said network element pairs in said second set of network element pairs; and

compute a computed network performance metric between a second network element pair using said measured network performance metric, wherein said second network element pair comprises a first network element and a second network element of said network element pair in said first set of network element pairs.

1 14. The computer system of claim 13, wherein said computed network
2 performance metric is computed using a relationship between said first and said
3 second network element pairs.

1 15. The computer system of claim 13, wherein said computer code
2 configured to cause said processor to order said first plurality of network element
3 pairs, is further configured to cause said processor to:
4 identify a plurality of network elements, wherein each one of said plurality of
5 network elements is one of a network element pair in said first set of
6 network element pairs;
7 assign one of a plurality of preferences to each one of said plurality of network
8 elements; and
9 sort said network element pairs in said first set of network element pairs based
10 on said plurality of preferences.

1 16. The computer system of claim 15, wherein said computer code
2 configured to cause said processor to sort said network element pairs in said first set
3 of network element pairs based on said plurality of preferences, is further configured
4 to cause said processor to:
5 for each one of said network element pairs in said first set of network element
6 pairs, swap a first network element and a second network element in
7 said each one of said network element pairs in said first set of network
8 element pairs, if a preference of said first network element in said each
9 one of said network element pairs in said first set of network element
10 pairs is less than a preference of said second network element in said
11 each one of said network element pairs in said first set of network
12 element pairs;
13 sort said network element pairs in said first set of network element pairs based
14 on a preference of a present first network element of said each one of

09638480, 073400

15 said network element pairs in said first set of network element pairs;
16 and
17 sort said network element pairs in said first set of network element pairs based
18 on a preference of a present second network element of said each one
19 of said network element pairs in said first set of network element pairs.

1 17. The computer system of claim 13, wherein said computer code is
2 further configured to cause said processor to:
3 form a first matrix, wherein each row in said first matrix corresponds to a
4 network element pair of said first set of network element pairs; and
5 determine a set of independent rows of said first matrix.

1 18. The computer system of claim 17, wherein said set of independent
2 rows of said first matrix is a maximal set of independent rows of said first matrix.

1 19. The computer system of claim 17, wherein said computer code
2 configured to cause said processor to form said second set of network element pairs, is
3 further configured to cause said processor to:
4 include independent network element pairs in said second set of network
5 element pairs, wherein
6 said independent network element pairs are ones of said network
7 element pairs in said first set of network element pairs
8 corresponding to rows of said first matrix in said set of
9 independent rows of said first matrix.

1 20. The computer system of claim 13, wherein said computer code is
2 further configured to cause said processor to:
3 form a delay components vector;
4 form a first matrix, wherein
5 said first matrix describes a relationship between said delay
6 components vector and a delay between each of said network
7 element pairs of said first set of network element pairs,

8 each row in said first matrix corresponds to a network element pair of
 9 said first set of network element pairs; and
 10 determine a set of independent rows of said first matrix, wherein said forming
 11 said second set of network element pairs comprises
 12 including independent network element pairs in said second set of
 13 network element pairs, and
 14 said independent network element pairs are ones of said network
 15 element pairs in said first set of network element pairs
 16 corresponding to rows of said first matrix in said set of
 17 independent rows of said first matrix.

1 21. The computer system of claim 20, wherein said delay components
 2 vector comprises:
 3 a representation of a delay within each network element of each network
 4 element pair of said first set of network element pairs for said each
 5 network element pair of said first set of network element pairs, and
 6 a representation of a delay between network elements of said each network
 7 element pair of said first set of network element pairs for said each
 8 network element pair of said first set of network element pairs.

1 22. The computer system of claim 20, wherein said computer code is
 2 further configured to cause said processor to:
 3 form a second matrix, wherein
 4 said second matrix describes a relationship between a plurality of
 5 independent delays and a non-independent delay,
 6 said plurality of independent delays comprise a delay between network
 7 elements in each network element pair of said second set of
 8 network element pairs, and
 9 said non-independent delay comprises a delay between network
 10 elements in a network element pair of said first set of network

17 one of said network element pairs in said second set of network
18 element pairs; and
19 a fifth set of instructions, executable on said computer system, configured to
20 compute a computed network performance metric between a second
21 network element pair using said measured network performance
22 metric, wherein said second network element pair comprises a first
23 network element and a second network element of said network
24 element pair in said first set of network element pairs.

1 26. The computer program product of claim 25, wherein fifth set of
2 instructions cause said computer system to compute said computed network
3 performance metric using a relationship between said first and said second network
4 element pairs.

1 27. The computer program product of claim 25, wherein said second set of
2 instructions comprises:
3 a sixth set of instructions, executable on said computer system, configured to
4 identify a plurality of network elements, wherein each one of said
5 plurality of network elements is one of a network element pair in said
6 first set of network element pairs;
7 a seventh set of instructions, executable on said computer system, configured
8 to identify a plurality of network elements, wherein each one of said
9 plurality of network elements is one of a network element pair in said
10 first set of network element pairs; and
11 a eighth set of instructions, executable on said computer system, configured to
12 identify a plurality of network elements, wherein each one of said
13 plurality of network elements is one of a network element pair in said
14 first set of network element pairs.

1 28. The computer program product of claim 27, wherein said eighth set of
2 instructions comprises:

3 a first sub-set of instructions, executable on said computer system, configured
4 to, for each one of said network element pairs in said first set of
5 network element pairs, swap a first network element and a second
6 network element in said each one of said network element pairs in said
7 first set of network element pairs, if a preference of said first network
8 element in said each one of said network element pairs in said first set
9 of network element pairs is less than a preference of said second
10 network element in said each one of said network element pairs in said
11 first set of network element pairs;

12 a second sub-set of instructions, executable on said computer system,
13 configured to sort said network element pairs in said first set of
14 network element pairs based on a preference of a present first network
15 element of said each one of said network element pairs in said first set
16 of network element pairs; and

17 an third sub-set of instructions, executable on said computer system,
18 configured to sort said network element pairs in said first set of
19 network element pairs based on a preference of a present second
20 network element of said each one of said network element pairs in said
21 first set of network element pairs.

1 29. The computer program product of claim 25, further comprising:
2 a sixth set of instructions, executable on said computer system, configured to
3 form a first matrix, wherein each row in said first matrix corresponds
4 to a network element pair of said first set of network element pairs; and
5 a seventh set of instructions, executable on said computer system, configured
6 to determine a set of independent rows of said first matrix.

1 30. The computer program product of claim 29, wherein said set of
2 independent rows of said first matrix is a maximal set of independent rows of said
3 first matrix.

1 31. The computer program product of claim 29, wherein said third set of
2 instructions comprises:
3 a first sub-set of instructions, executable on said computer system, configured
4 to include independent network element pairs in said second set of
5 network element pairs, wherein
6 said independent network element pairs are ones of said network
7 element pairs in said first set of network element pairs
8 corresponding to rows of said first matrix in said set of
9 independent rows of said first matrix.

1 32. The computer program product of claim 25, further comprising:
2 a sixth set of instructions, executable on said computer system, configured to
3 form a delay components vector;
4 a seventh set of instructions, executable on said computer system, configured
5 to form a first matrix, wherein
6 said first matrix describes a relationship between said delay
7 components vector and a delay between each of said network
8 element pairs of said first set of network element pairs,
9 each row in said first matrix corresponds to a network element pair of
10 said first set of network element pairs; and
11 a eighth set of instructions, executable on said computer system, configured to
12 determine a set of independent rows of said first matrix, wherein said
13 forming said second set of network element pairs comprises
14 including independent network element pairs in said second set of
15 network element pairs, and

09633130 073100

16 said independent network element pairs are ones of said network
 17 element pairs in said first set of network element pairs
 18 corresponding to rows of said first matrix in said set of
 19 independent rows of said first matrix.

1 33. The computer program product of claim 32, wherein said delay
 2 components vector comprises:
 3 a representation of a delay within each network element of each network
 4 element pair of said first set of network element pairs for said each
 5 network element pair of said first set of network element pairs, and
 6 a representation of a delay between network elements of said each network
 7 element pair of said first set of network element pairs for said each
 8 network element pair of said first set of network element pairs.

1 34. The computer program product of claim 32, further comprising:
 2 a ninth set of instructions, executable on said computer system, configured to
 3 form a second matrix, wherein
 4 said second matrix describes a relationship between a plurality of
 5 independent delays and a non-independent delay,
 6 said plurality of independent delays comprise a delay between network
 7 elements in each network element pair of said second set of
 8 network element pairs, and
 9 said non-independent delay comprises a delay between network
 10 elements in a network element pair of said first set of network
 11 element pairs that is not in said second set of network element
 12 pairs.

1 35. The computer program product of claim 34, wherein said ninth set of
2 instructions comprises:
3 a first sub-set of instructions, executable on said computer system, configured
4 to perform a Gaussian elimination using said first and said second
5 matrices.

1 36. The computer program product of claim 34, wherein said fifth set of
2 instructions comprises:
3 a first sub-set of instructions, executable on said computer system, configured
4 to multiply said measured performance metric by an element of said
5 second matrix.

1 37. A computer system comprising:
2 a network interface, coupled to said processor and to a network, wherein said
3 network comprises a plurality of network elements and each one of
4 said network elements is coupled to at least one other of said network
5 elements by at least one of a plurality of links;
6 means for forming a first set of network element pairs, said first set of network
7 element pairs comprising a plurality of pairs of said network elements;
8 means for ordering a first plurality of network element pairs comprising ones
9 of network element pairs in said first set of network element pairs;
10 means for forming a second set of network element pairs, wherein said second
11 set of network element pairs comprises ones of said network element
12 pairs in said first set of network element pairs
13 means for measuring a measured network performance metric between a first
14 network element pair, wherein said first network element pair
15 comprises a first network element and a second network element of
16 one of said network element pairs in said second set of network
17 element pairs; and

18 means for computing a computed network performance metric between a
 19 second network element pair using said measured network
 20 performance metric, wherein said second network element pair
 21 comprises a first network element and a second network element of
 22 said network element pair in said first set of network element pairs.

1 38. The method of claim 37, wherein said computed network performance
 2 metric is computed using a relationship between said first and said second network
 3 element pairs.

1 39. The computer system of claim 37, further comprising:
 2 means for identifying a plurality of network elements, wherein each one of
 3 said plurality of network elements is one of a network element pair in
 4 said first set of network element pairs;
 5 means for assigning one of a plurality of preferences to each one of said
 6 plurality of network elements; and
 7 means for sorting said network element pairs in said first set of network
 8 element pairs based on said plurality of preferences.

1 40. The computer system of claim 39, wherein said sorting means
 2 comprises:
 3 means, for each one of said network element pairs in said first set of network
 4 element pairs, for swapping a first network element and a second
 5 network element in said each one of said network element pairs in said
 6 first set of network element pairs, if a preference of said first network
 7 element in said each one of said network element pairs in said first set
 8 of network element pairs is less than a preference of said second
 9 network element in said each one of said network element pairs in said
 10 first set of network element pairs.
 11 means for sorting said network element pairs in said first set of network
 12 element pairs based on a preference of a present first network element

13 of said each one of said network element pairs in said first set of
 14 network element pairs; and
 15 means for sorting said network element pairs in said first set of network
 16 element pairs based on a preference of a present second network
 17 element of said each one of said network element pairs in said first set
 18 of network element pairs.

1 41. The computer system of claim 37, further comprising:
 2 means for forming a first matrix, wherein each row in said first matrix
 3 corresponds to a network element pair of said first set of network
 4 element pairs; and
 5 means for determining a set of independent rows of said first matrix.

1 42. The computer system of claim 41, wherein said set of independent
 2 rows of said first matrix is a maximal set of independent rows of said first matrix.

1 43. The computer system of claim 41, wherein said means for forming said
 2 second set of network element pairs comprises:
 3 means for including independent network element pairs in said second set of
 4 network element pairs, wherein
 5 said independent network element pairs are ones of said network
 6 element pairs in said first set of network element pairs
 7 corresponding to rows of said first matrix in said set of
 8 independent rows of said first matrix.

1 44. The computer system of claim 37, further comprising:
 2 means for forming a delay components vector;
 3 means for forming a first matrix, wherein
 4 said first matrix describes a relationship between said delay
 5 components vector and a delay between each of said network
 6 element pairs of said first set of network element pairs,

09633180 "03400

7 each row in said first matrix corresponds to a network element pair of
8 said first set of network element pairs; and
9 means for determining a set of independent rows of said first matrix, wherein
10 said forming said second set of network element pairs comprises
11 including independent network element pairs in said second set of
12 network element pairs, and
13 said independent network element pairs are ones of said network
14 element pairs in said first set of network element pairs
15 corresponding to rows of said first matrix in said set of
16 independent rows of said first matrix.

1 45. The computer system of claim 44, wherein said delay components
2 vector comprises:
3 a representation of a delay within each network element of each network
4 element pair of said first set of network element pairs for said each
5 network element pair of said first set of network element pairs, and
6 a representation of a delay between network elements of said each network
7 element pair of said first set of network element pairs for said each
8 network element pair of said first set of network element pairs.

1 46. The method of claim 44, further comprising:
2 means for forming a second matrix, wherein
3 said second matrix describes a relationship between a plurality of
4 independent delays and a non-independent delay,
5 said plurality of independent delays comprise a delay between network
6 elements in each network element pair of said second set of
7 network element pairs, and
8 said non-independent delay comprises a delay between network
9 elements in a network element pair of said first set of network
10 element pairs that is not in said second set of network element
11 pairs.

00638180 073100

1 48. The computer system of claim 46, wherein said means for computing
2 said computed network performance metric comprises:
3 means for multiplying said measured performance metric by an element of
4 said second matrix.

Table 1. Demographic characteristics of the study population	
Age (years)	50.0 ± 10.0
Gender	
Male	50.0%
Female	50.0%
Marital status	
Married	80.0%
Single	20.0%
Education level	
High school or less	30.0%
College or more	70.0%
Occupation	
Professional	40.0%
Managerial	30.0%
Technical	20.0%
Service	10.0%
Unemployed	0.0%
Income (USD/month)	
< 1000	10.0%
1000-2000	30.0%
2000-3000	40.0%
> 3000	20.0%
Health insurance	
Yes	90.0%
No	10.0%
Smoking status	
Smoker	20.0%
Non-smoker	80.0%
Alcohol consumption	
Regular	10.0%
Occasional	30.0%
Never	60.0%
Family size	
1-2	30.0%
3-4	40.0%
5-6	20.0%
> 6	10.0%

**METHOD AND APPARATUS FOR ESTIMATING DELAY AND JITTER
BETWEEN MANY NETWORK ROUTERS USING MEASUREMENTS
BETWEEN A PREFERRED SET OF ROUTERS**

Jain, Bijendra N.
McCloghrie, Keith

5

ABSTRACT

A method and apparatus for determining a network performance metric in a network is described. The network includes a number of network elements and a number of links. Each of the network elements is coupled to at least one other of the network elements by at least one of the links. The method includes forming a first set of network element pairs, ordering a first number of network element pairs, forming a second set of network element pairs, measuring a measured network performance metric between a first network element pair and computing a computed network performance metric. The first set of network element pairs include a number of pairs of the network elements. The ordering of a first number of network element pairs includes network element pairs in the first set of network element pairs. The second set of network element pairs includes network element pairs in the first set of network element pairs. The measurement is taken between a first network element pair. The first network element pair includes a first network element and a second network element of one of the network element pairs in the second set of network element pairs. The computed network performance metric is computed between a second network element pair using the measured network performance metric. The second network element pair includes a first network element and a second network element of the network element pair in the first set of network element pairs.

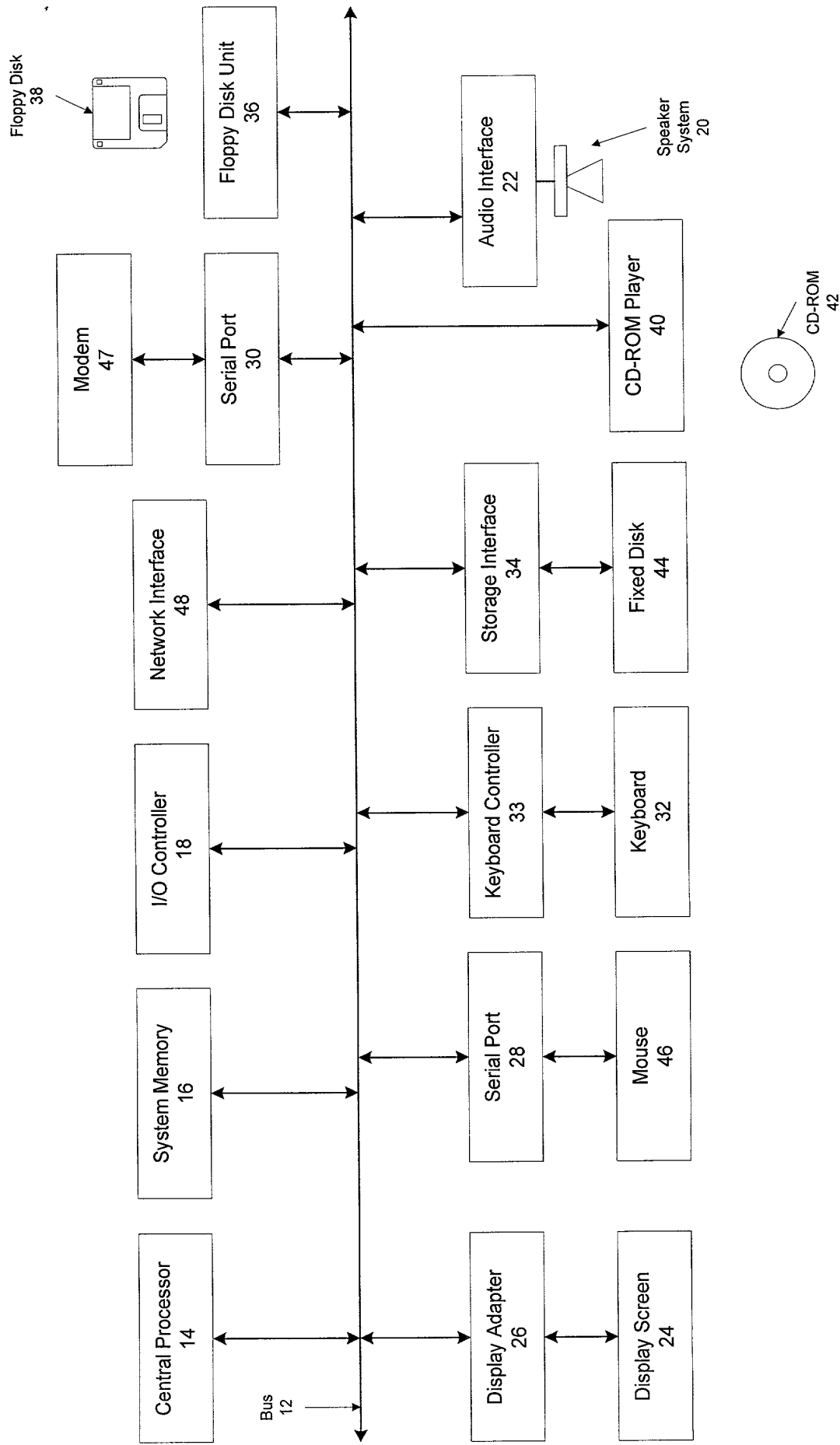


FIG. 1A

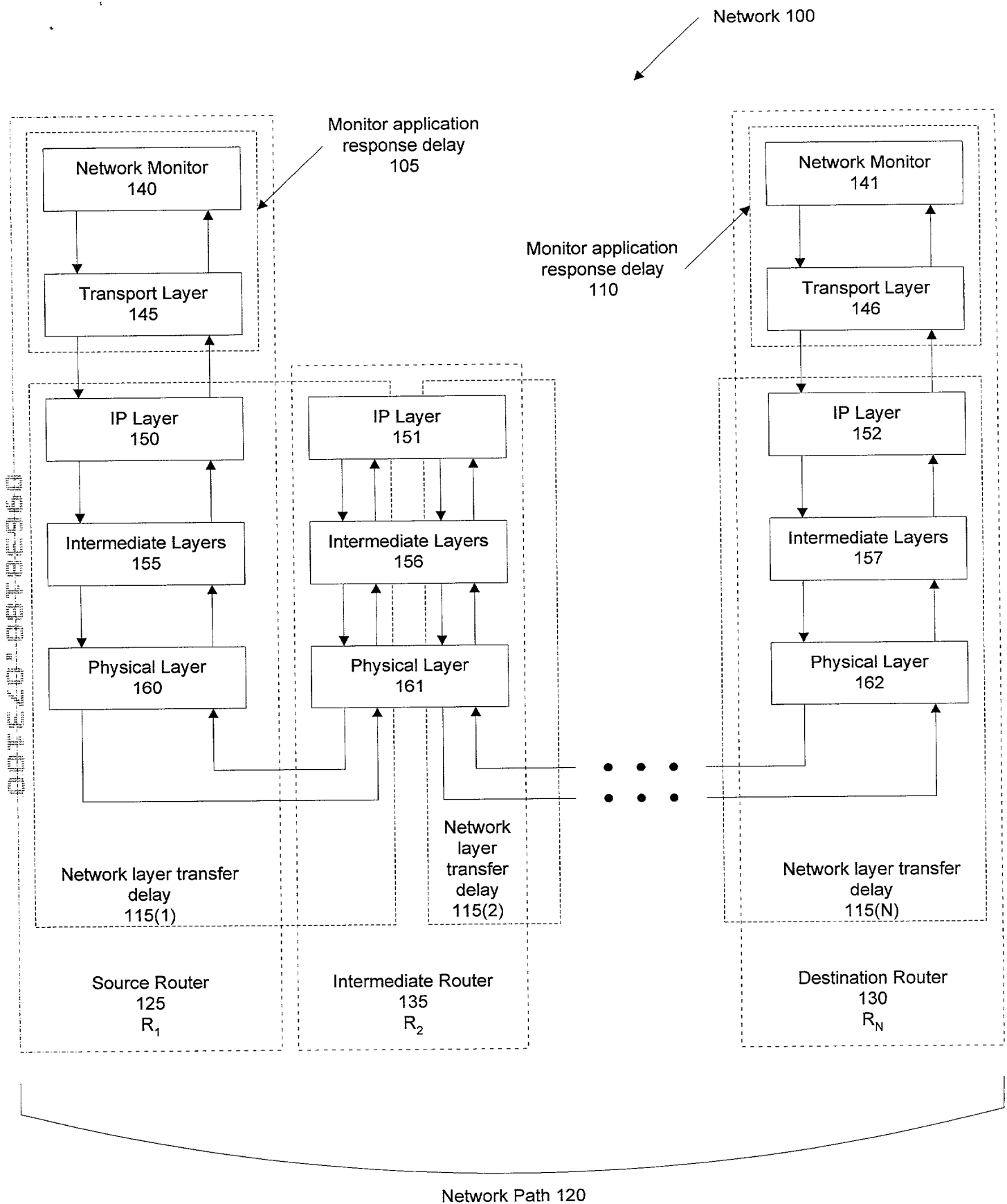


Fig. 1B

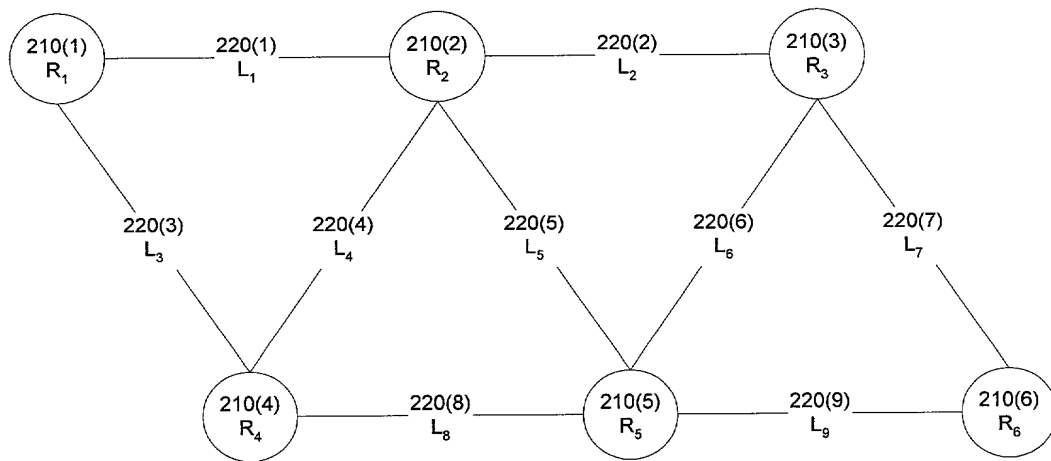


Fig. 2

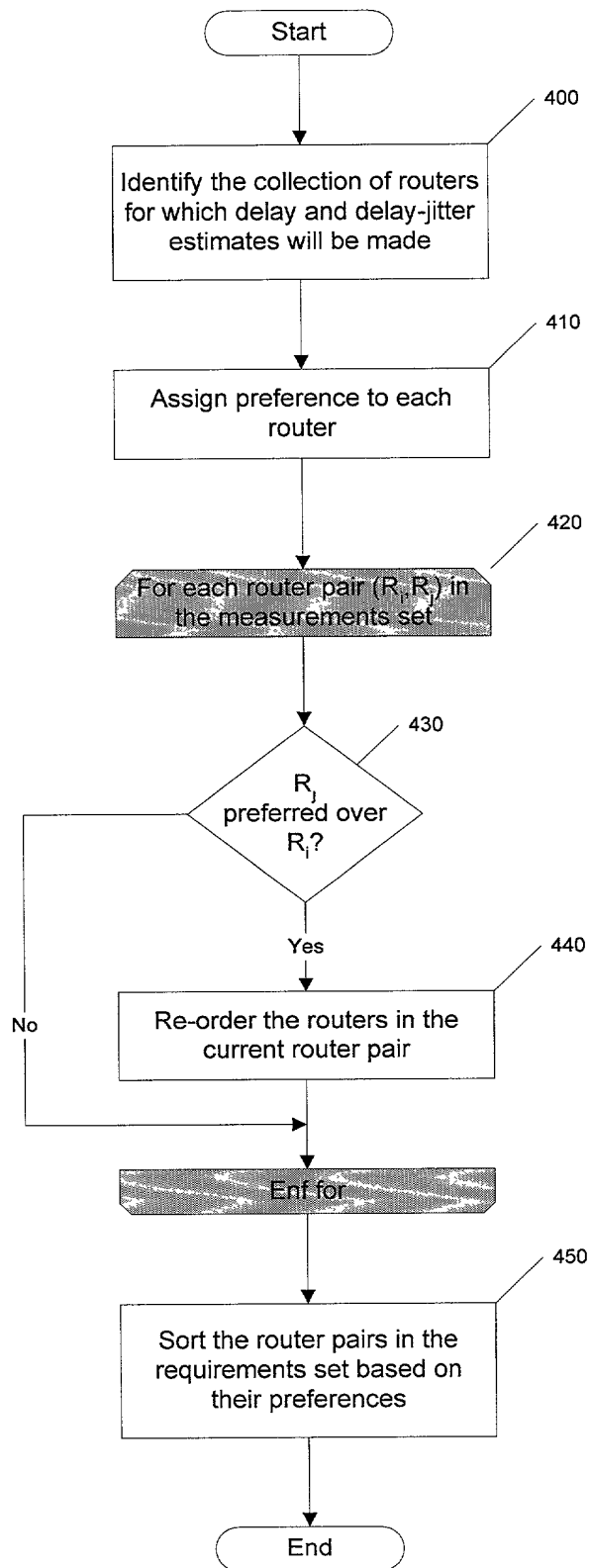


Fig. 4

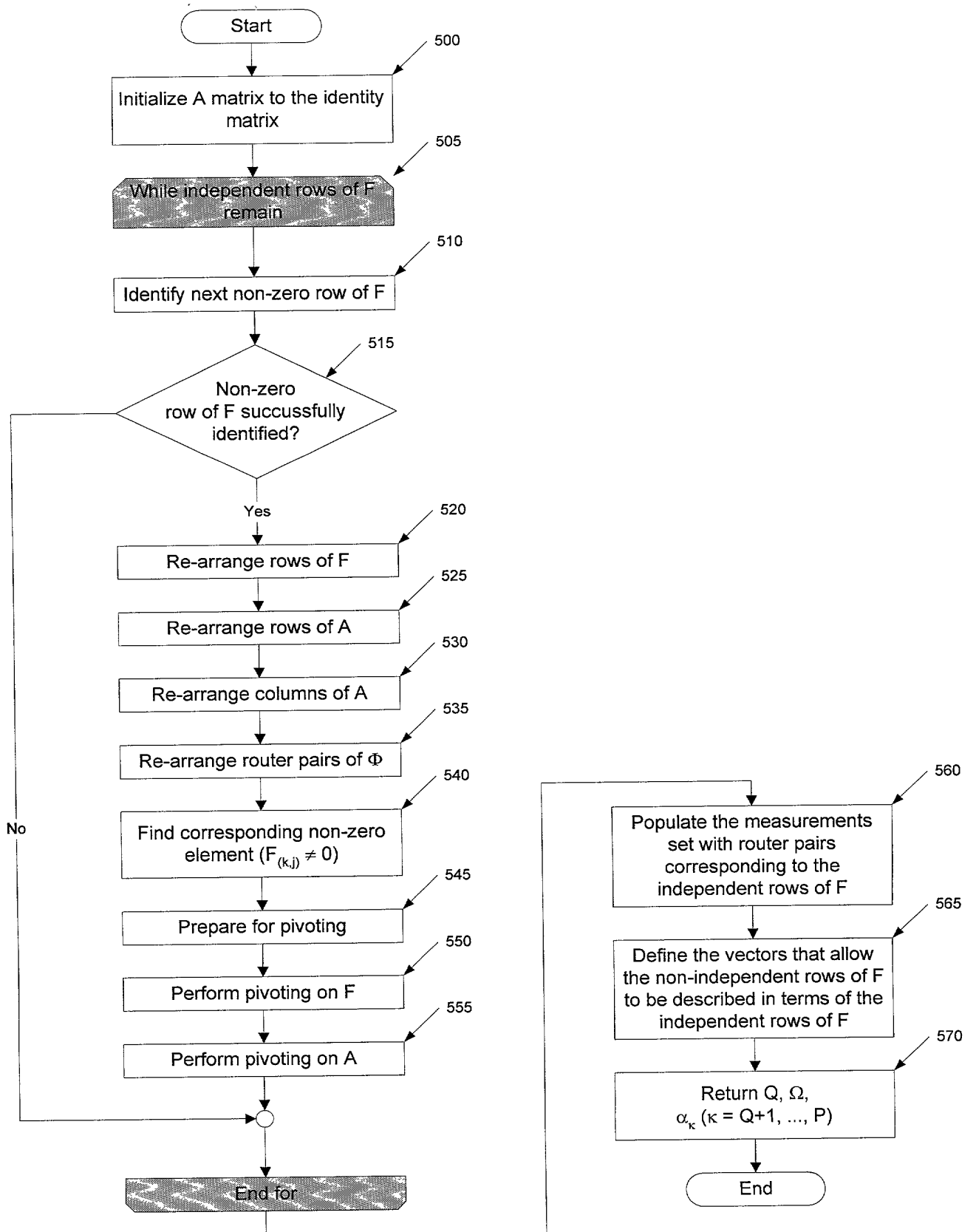


Fig. 5

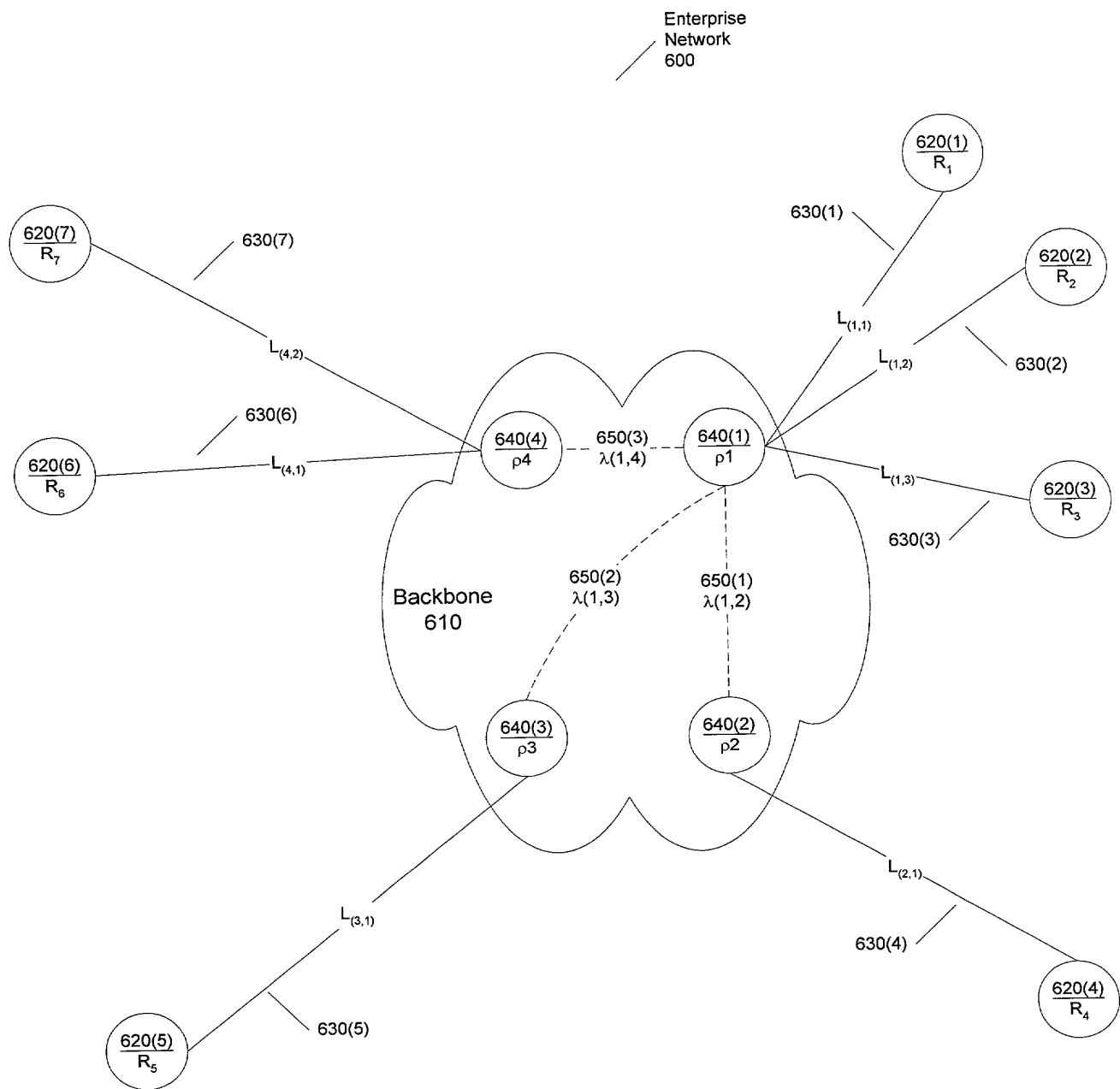


Fig. 6

**DECLARATION FOR PATENT APPLICATION
AND POWER OF ATTORNEY**

As a below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below adjacent to my name.

I believe I am the original, first and sole inventor (if only one name is listed below) or an original, first and joint inventor (if plural names are listed below) of subject matter (process, machine, manufacture, or composition of matter, or an improvement thereof) which is claimed and for which a patent is sought by way of the application entitled

**Method And Apparatus For Estimating Delay And Jitter
Between Many Network Routers Using Measurements
Between A Preferred Set of Routers**

which (check) ☒ is attached hereto.
☐ and is amended by the Preliminary Amendment attached hereto.
☐ was filed on _____ as Application Serial No. _____
☐ and was amended on ____ (if applicable).

I hereby state that I have reviewed and understand the contents of the above identified specification, including the claims, as amended by any amendment referred to above.

I acknowledge the duty to disclose information, which is material to patentability as defined in Title 37, Code of Federal Regulations, § 1.56.

I hereby claim foreign priority benefits under Title 35, United States Code, § 119(a)-(d) of any foreign application(s) for patent or inventor's certificate or any PCT international application(s) designating at least one country other than the United States of America listed below and have also identified below any foreign application(s) for patent or inventor's certificate or any PCT international application(s) designating at least one country other than the United States of America filed by me on the same subject matter having a filing date before that of the application(s) of which priority is claimed:

Prior Foreign Application(s)			Priority Claimed	
Number	Country	Day/Month/Year Filed	Yes	No
N/A			<input type="checkbox"/>	<input type="checkbox"/>

I hereby claim the benefit under Title 35, United States Code, § 119(e) of any United States provisional application(s) listed below:

Provisional Application Number	Filing Date
N/A	

I hereby claim the benefit under Title 35, United States Code, § 120 of any United States application(s) or PCT international application(s) designating the United States of America listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior application(s) in the manner provided by the first paragraph of Title 35, United States Code, § 112, I acknowledge the duty to disclose information, which is material to patentability as defined in Title 37, Code of Federal Regulations, § 1.56, which became available between the filing date of the prior application(s) and the national or PCT international filing date of this application:

Application Serial No.	Filing Date	Status (patented, pending, abandoned)
N/A		

I hereby appoint the following attorney(s) and/or agent(s) to prosecute this application and to transact all business in the United States Patent and Trademark Office connected therewith:

Alan H. MacPherson (24,423); Brian D. Ogonowsky (31,988); David W. Heid (25,875); Norman R. Klivans (33,003); Edward C. Kwok (33,938); David E. Steuber (25,557); Michael Shenker (34,250); Stephen A. Terrile (32,946); Peter H. Kang (40,350); Ronald J. Meetin (29,089); Ken John Koestner (33,004); Omkar K. Suryadevara (36,320); David T. Millers (37,396); Michael P. Adams (34,763); Robert B. Morrill (43,817); James E. Parsons (34,691); Philip W. Woo (39,880); Emily Haliday (38,903); Tom Hunter (38,498); Michael J. Halbert (40,633); Gary J. Edwards (41,008); Daniel P. Stewart (41,332); John T. Winburn (26,822); Tom Chen (42,406); Fabio E. Marino (43,339); Don C. Lawrence (31,975); Marc R. Ascolese (42,268); Carmen C. Cook (42,433); David G. Dolezal (41,711); Roberta P. Saxon (43,087); Mary Jo Bertani (42,321); Dale R. Cook (42,434); Sam G. Campbell (42,381); Matthew J. Brigham (44,047); Hugh H. Matsubayashi (43,779); Patrick D. Benedicto (40,909); T.J. Singh (39,535); Shireen Irani Bacon (40,494); Rory G. Bens (44,028); George Wolken, Jr. (30,441); John A. Odozynski (28,769); Cameron K. Kerrigan (44,826); Paul E. Lewkowicz (44,870); Theodore P. Lopez (44,881); Mayankkumar M. Dixit (44,064); Eric Stephenson (38,321); Christopher Allenby (45,906); David C. Hsia (46,235); Mark J. Rozman (42,117); Margaret M. Kelton (44,182); Do Te Kim (46,231); Alex Chen (45,591); Monique M. Heyninck (44,763); Gregory J. Michelson (44,940); Jonathan Geld (44,702); Emmanuel Rivera (45,760); Jason FarHadian (42,523); Matthew J. Spark (43,453); and Elaine H. Lo (41,158).

Please address all correspondence and telephone calls to:

Sam Campbell
Attorney for Applicant(s)
SKJERN MORRILL MacPHERSON LLP
25 Metro Drive, Suite 700
San Jose, California 95110-1349

Telephone: 512-794-3600
Facsimile: 512-794-3601

I declare that all statements made herein of my own knowledge are true, all statements made herein on information and belief are believed to be true, and all statements made herein are made with the knowledge that whoever, in any matter within the jurisdiction of the Patent and Trademark Office, knowingly and willfully falsifies, conceals, or covers up by any trick, scheme, or device a material fact, or makes any false, fictitious or fraudulent statements or representations, or makes or uses any false writing or document knowing the same to contain any false, fictitious or fraudulent statement or entry, shall be subject to the penalties including fine or imprisonment or both as set forth under 18 U.S.C. 1001, and that violations of this paragraph may jeopardize the validity of the application or this document, or the validity or enforceability of any patent, trademark registration, or certificate resulting therefrom.

